

Modelo para estimación de la calidad de un Web Service

María Pérez, Luis E. Mendoza, Anna C. Grimán,

Universidad Simón Bolívar, Departamento de Procesos y Sistemas, LISI

Caracas – Venezuela, Apartado Postal 89000, Z.P. 1080-A

{movalles, lmendoza, agriman}@usb.ve

Resumen

Los Web Services ofrecen una alternativa de *software* independiente en cuanto a la plataforma, basada en estándares para la integración de aplicaciones, la automatización de procesos de negocio y la publicación de la información de diversas fuentes. La calidad en Web Service es vital para una organización que busca apalancarse en la integración. Contar con una infraestructura integrada, segura, escalable y disponible disminuye costos y permite compartir información de manera confiable. Una forma de responder esta necesidad es valorando el nivel de calidad del Web Service, por lo cual se hace necesario desarrollar un *Modelo de Estimación de la Calidad* con el uso de criterios de evaluación considerados como estándares, aplicados al caso específico de Web Service. Este modelo responde a la necesidad de disponer de una herramienta que estime la calidad de estos productos en una manera específica. Las empresas podrán contar así con un mecanismo confiable que permita determinar los aspectos de calidad relevantes de los productos desarrollados. Este modelo fue evaluado a través de un Análisis de Características en dos estudios de caso, arrojando que el mismo es pertinente, completo, adecuado y preciso.

Palabras Claves: Modelo de Calidad, Web Service

Abstract

Web Services offer an alternative of independent software as far as the platform, based on standards for the integration of applications, the business processes automatization and the publication of the information of diverse sources. A high-quality Web Service is medullar for an organization demanding integration support. Counting with an integrated, safe, scaleable and available infrastructure diminishes costs and allows sharing information in a reliable way. A form to respond this necessity is valuing the quality level of the Web Service. In order to accomplish this, a *Model of Quality Estimation is developed by using* standard evaluation criteria, applied to the specific case of Web Service. This model responds to the necessity to have a tool that considers the quality of these products in a specific way. Thus, companies will be able to count on a reliable mechanism that allows determining the excellent aspects of quality of developed products. This model was evaluated through Feature Analysis in two case studies, concluding that it is pertinent, complete, suitable and precise.

Keywords: Quality Model, Web Service

1. INTRODUCCIÓN

La estimación de la calidad de software es, en general, un proceso difícil de llevar a cabo ya que requiere de un compromiso por parte de los agentes externos a la organización a evaluar y por parte de sus miembros, los cuales orientan la evaluación según sus necesidades. En particular, este tópico empleado en tecnologías novedosas como son los Web Service (WS) se convierten en un reto. En este sentido, la presente investigación propone una adecuación del *Modelo Sistémico de Calidad* (MOSCA) [18], para que éste sea aplicable a la nueva tecnología que impulsa a las empresas: los WS. Los WS ofrecen una alternativa de software independiente en cuanto a la plataforma, basada en estándares para la integración de aplicaciones, la automatización de procesos de negocio y la publicación de información de diversas fuentes e intercambio empresarial. Lo que lleva a preguntarse es si en realidad se están desarrollando de manera provechosa para las organizaciones. Poseer un WS de calidad es de importancia para una organización ya que se benefician las relaciones de negocio contribuyendo tanto a la integración interna como la inter-empresarial; al asegurar una infraestructura integrada, segura, escalable y disponible se disminuye el costo de propiedad y permite

compartir información de manera confiable.

Para ello se propone un Modelo de Estimación de la Calidad de Web Service basado en MOSCA y en el estándar Calidad de Servicio (Quality of Service, QoS), como una solución al problema de no disponer de una herramienta que valore la Calidad de los WS. El principal aporte de este modelo es que parte de una base ya probada (MOSCA), el cual a su vez se inspira en estándares de calidad y propone métricas, para luego incorporar los aspectos propios de la calidad de Web Service y QoS, donde éste último es también una fuente de requerimientos importante íntimamente relacionados al mundo de la Ingeniería Web y las aplicaciones en Internet. En consecuencia, las empresas que aplican esta tecnología podrán conocer los atributos de calidad que están presentes en su sistema (confiabilidad y seguridad, entre otros) y las recomendaciones oportunas que favorecen su desarrollo. Como logro adicional, se evaluó el modelo propuesto con dos estudios de caso, encontrándose que el mismo es pertinente, completo, adecuado y preciso.

A parte de la Introducción y las Conclusiones, este artículo está conformado por cuatro secciones adicionales, la primera describe los antecedentes de la investigación, en la segunda se identifican las características de calidad para los WS, en la tercera se propone el modelo de calidad y en la cuarta se describe la evaluación del modelo con los resultados más relevantes.

2. MARCO REFERENCIAL

La calidad según ISO/IEC 9126 [9] es la totalidad de las características de una entidad que refieren su capacidad de satisfacer necesidades establecidas o implícitas. La revisión bibliográfica [4, 5, 8, 9, 10, 11, 26] refleja la tendencia de formular modelos que especifiquen la calidad del software. Tomando en cuenta la calidad del producto y la calidad del proceso, se desarrolló MOSCA [18], que integra el modelo de calidad del producto [20] y el modelo de calidad del proceso de desarrollo [22], y está soportado por los conceptos de calidad total sistémica [2, 21]. En cuanto a la calidad del Producto, MOSCA plantea, sobre la base de las 6 características de calidad del estándar internacional ISO/IEC 9126 [9], un conjunto de categorías, características y métricas asociadas, que especifican la calidad de un producto de software con un enfoque sistémico y hacen del modelo un instrumento de medición de valor ya que cubre todos los aspectos imprescindibles para medir directamente la calidad del producto de software.

La literatura [6, 13, 16, 19, 25] precisa que los WS son aplicaciones multiplataforma basadas en componentes interoperables que utilizan estándares de protocolos de Internet; además, permiten crear una única puerta de entrada a la organización exponiendo las funcionalidades del negocio sobre la Web, facilitando así la integración del mismo.

La Figura 1 muestra la arquitectura conceptual orientada a servicio (Service Oriented Architecture, SOA) usando las tecnologías: Protocolo de Acceso de Objeto Simple (Simple Object Access Protocol, SOAP), Lenguaje de Descripción de los Servicios Web (Web Services Description Language, WSDL) e Integración, Descubrimiento y Descripción Universal (Universal Description, Discovery and Integration, UDDI) y mostrando los servicios que la conforman en el marco de los WS.

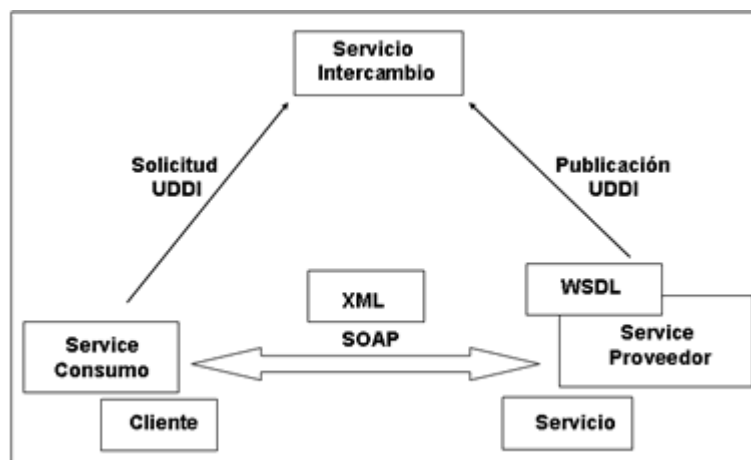


Figura 1. Vista de implementación de SOA con las tecnologías SOAP, WSDL, y UDDI [25].

Según IEEE, QoS para aplicaciones que deben comunicarse en tiempo real, es el conjunto de características cualitativas y cuantitativas de un sistema multimedia distribuido, que son necesarias para lograr la

funcionalidad requerida en una aplicación [7]. En este sentido, QoS determina la usabilidad y utilidad del servicio, influenciando en la popularidad del mismo. Con la extensa propagación de los WS, QoS se ha convertido en un factor significativo para determinar el éxito que pueda alcanzar un proveedor de servicio sobre otros [14]. Aplicar QoS en Internet es un reto crítico y significativo debido a la naturaleza dinámica e impredecible del Web.. QoS cubre un rango de técnicas que armoniza las necesidades de quien requiere el servicio con quien lo provee, basándose en los recursos de red disponibles [14]. A través de QoS se puede hacer referencia a las propiedades no funcionales del WS, tales como rendimiento, fiabilidad, disponibilidad y seguridad [14], las cuales representan los requerimientos de calidad necesarios para alcanzar la funcionalidad esperada del mismo.

3. WEB SERVICES Y CARACTERÍSTICAS DE CALIDAD

Los WS pueden tener varias formas de clasificarse. Los autores conciben diferentes tipos de WS que no se contradicen entre sí. Según la funcionalidad de los Web services: Canta y sus colegas [3] dividen a los WS en dos categorías: WS Programáticos (Programmatic Web Services, PWS) y WS Interactivos (Interactive Web Services, IWS); los primeros encapsulan la funcionalidad en la capa de negocios de las aplicaciones, mientras que los segundos la encapsulan en la interfaz de usuario o la capa de presentación. Los PWS extienden la capa lógica de negocios de una aplicación. [3]. En los WS la característica de mantenibilidad se ve reflejada a través de la simpleza de las operaciones, que permiten la facilidad de cambio, análisis y pruebas; además propicia funcionalidad gracias a la posibilidad que posee el suscriptor de adecuar dichas funciones a sus requerimientos específicos. Los IWS exponen una interfaz de usuario de aplicación. Como los IWS se manejan a través de la interfaz del usuario, la característica de usabilidad se ve reflejada en ellos.

Según las necesidades del negocio. La empresa Microsoft clasifica a los WS según la funcionalidad que el negocio desee darle al mismo:

3.1 Los WS Orientados a datos (Web Services Data Centric, WSDC) que son útiles en situaciones donde deben actualizarse datos que son alterados frecuentemente. [15]Se puede deducir que los WSDC propician la característica de eficiencia al negocio; por lo que las organizaciones que requieren de la adecuada utilización del tiempo y los recursos, ven conveniente la aplicación de este tipo de WS para obtener los beneficios esperados.

3.2 Los WS de Colaboración (Web Service Collaboration, WSC) son aquellos que permiten la colaboración entre todos los involucrados del negocio [15]. Estos WS suministran la característica de interoperabilidad al negocio.

3.3. Los WS para Análisis (Web Service Analysis, WSA) son aquellos que reciben informes de varias empresas filiales, los ejecuta, consolida y entrega los resultados deseados [15]. Por lo tanto, este tipo de WS impulsa la característica de usabilidad del negocio.

3.4 Los WS de Alertas (Web Services Alert, WSAL) manejan situaciones de alerta más eficientemente, ya que se puede enviar mensajes de aviso a un dispositivo móvil (celular, pda, etc) para que se pueda actuar de inmediato [15].La característica fundamental que propicia al negocio este tipo de WS, es la eficiencia.

Los requerimientos de los usuarios establecen los criterios para seleccionar el tipo de WS que se ha de aplicar. Estos requerimientos exigen características de calidad que el usuario espera del WS. La tabla 1 resume la clasificación planteada anteriormente junto con la(s) característica(s) de calidad asociada(s) a cada una de ellas y traducidas al estándar ISO 9126 según [12].

Tipo de WS	Característica de calidad del negocio	Según ISO/EIC 9126
PWS	Mantenibilidad, Funcionalidad	Mantenibilidad, Funcionalidad
IWS	Usabilidad	Usabilidad
WSDC	Eficiencia	Eficiencia
WSC	Interoperabilidad	Funcionalidad /Interoperabilidad
WSA	Usabilidad	Usabilidad
WSAL	Eficiencia	Eficiencia
QoS	Rendimiento, Fiabilidad, Seguridad	Eficiencia, Fiabilidad, Funcionalidad/Seguridad

Tabla 1. Características de Calidad asociadas al tipo de WS.

4 MODELO DE CALIDAD PARA WEB SERVICE

Para el desarrollo del producto de calidad aquí propuesto, se partió de MOSCA [18]. Este modelo fue desarrollado por el Laboratorio de Información y Sistemas de Información (LISI) de la Universidad Simón Bolívar, con el financiamiento del gobierno venezolano. Ha sido probado en más de seis estudios de caso

[23]. Se basa en un enfoque de calidad sistémica el cual contempla tanto la perspectiva producto (software) como la perspectiva proceso. Con base a esta experiencia y a su condición sistémica, se decidió hacer una adaptación del mismo para WS. Para ello se siguió la metodología propuesta por Rincón et al. [24]. Como consecuencia de esta adaptación, se utiliza sólo la perspectiva de producto, de ella se adaptaron tanto las categorías como las características; finalmente se propusieron 46 métricas. Adicionalmente, se compararon las características de calidad identificadas en la Tabla 1, concluyendo que se utilizarían las seis que presenta el MOSCA ya que son las mismas

A continuación se describe el modelo obtenido, el cual se puede apreciar en la Figura 2.

Nivel 0: Dimensiones. Aspectos Internos y Aspectos Contextuales del producto.

Nivel 1: Categorías. En el modelo se contemplan 6 categorías las cuales se describen en la Tabla 2.

Nivel 2: Características. Cada categoría tiene asociado un conjunto de características, las cuales definen las áreas claves que se deben satisfacer para lograr, asegurar y controlar la calidad de un WS. Las Tablas 3 a 8 muestran cada categoría con sus características asociadas. Las categorías de Usabilidad, Eficiencia y Fiabilidad, y sus correspondientes características, fueron adecuadas a WS.

Nivel 3: Métricas. Para cada característica se propone un conjunto de métricas que se formularon según el paradigma Meta-Pregunta-métrica (Goal Question Metric, GQM) de Basili [1], el cual fue útil al proporcionar un enfoque sistemático que permite ir desde unos requerimientos u objetivos, hasta las mediciones necesarias para valorar directamente los mismo. En este sentido, las metas pueden ser vistas como los atributos de calidad que se quieren medir. Por otra parte, para cada característica se orienta sobre la forma en que las respuestas deben ser dadas para poder determinar la evaluación final del WS. Las métricas propuestas se encuentran detalladas en la Tabla 9.

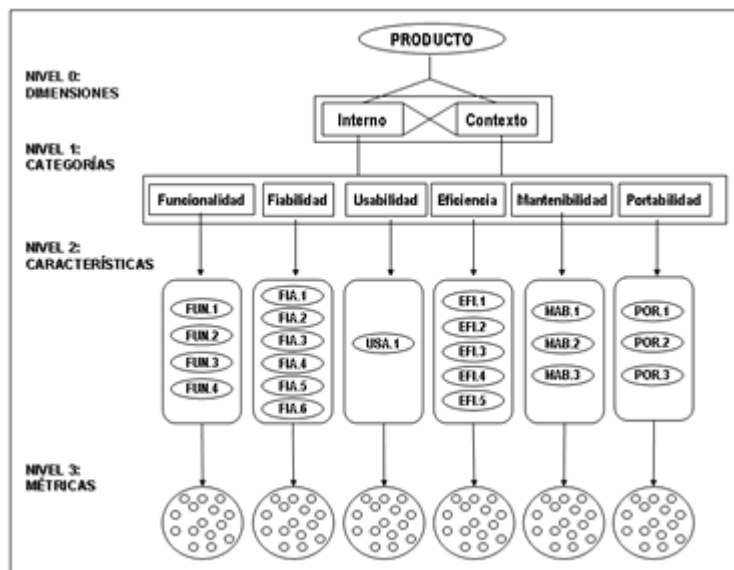


Figura 2. Representación gráfica de la especificación de MOSCA para WS.

Categorías	Definición
Funcionalidad (FUN)	Es la capacidad del Webservice para proveer las funciones que cumplan con las necesidades específicas o implícitas, cuando es utilizado bajo ciertas condiciones.
Fiabilidad (FIA)	La fiabilidad es la capacidad del Webservice para mantener un nivel especificado de rendimiento cuando es utilizado bajo condiciones especificadas.
Usabilidad (USA)	Esta categoría se refiere a la capacidad del Webservice para ser atractivo, entendido, aprendido y utilizado por el usuario bajo condiciones específicas.
Eficiencia (EFI)	Es la capacidad del Webservice para proveer un rendimiento apropiado, relativo a la cantidad de recursos utilizados, bajo condiciones específicas.
Mantenibilidad (MAB)	Es la capacidad del Webservice para ser modificado. Las modificaciones pueden incluir correcciones, mejoras o adaptaciones del software ante cambios del ambiente, en requerimientos y especificaciones funcionales.
Portabilidad (POR)	La portabilidad es la capacidad del Webservice para ser transferido de un ambiente a otro

Tabla 2. Categorías del Modelo de Calidad de Webservices.

Características	Descripción
FUN.1 Precisión	Capacidad del Webservice para proveer resultados o efectos correctos o convenientes. Esto incluye el grado de precisión de los valores calculados. Está relacionada con la correctitud en la ejecución de las transacciones.
FUN.2 Transacción	Capacidad de tratar una secuencia de actividades como una simple unidad de trabajo.
FUN.3 Seguridad	Capacidad del Webservice de proteger su información y datos así como la de controlar el acceso no autorizado al mismo
FUN.4 Interoperabilidad	Capacidad del Webservice para interactuar con uno o más sistemas (Propiciada por la tecnología – XML).

Tabla 3. Características del Modelo correspondiente a la categoría Funcionalidad.

Características	Descripción
EFI.1 Estandarización	Capacidad del Webservice para ajustarse a estándares, convenciones o regulaciones.
EFI.2 Comportamiento del tiempo	Capacidad del Webservice para proveer respuestas y tiempos de procesamiento apropiados en tiempo de ejecución bajo condiciones específicas.
EFI.3 Utilización de los recursos	Capacidad del Webservice para utilizar cantidades apropiadas de los recursos cuando el mismo ejecuta sus funciones bajo condiciones específicas.
EFI.4 Latencia	Tiempo transcurrido entre enviar una solicitud y recibir una respuesta.
EFI.5 Throughput	Capacidad del Webservice para representar el número de peticiones atendidas por un WS en un período de tiempo determinado.

Tabla 4. Características del Modelo correspondiente a la categoría Eficiencia.

Características	Descripción
FIA.1 Disponibilidad	Capacidad del producto del software de estar siempre presente y en estado operativo al momento de ejecutar una función en un período determinado, bajo condiciones específicas. Está relacionada con el tiempo de reparo.
FIA.2 Tiempo de Reparación	Representa el tiempo que toma reparar un servicio que ha fallado.
FIA.3 Accesibilidad	Según QoS, representa la capacidad de un servicio para atender una solicitud de un WS. Puede ocurrir que un WS este disponible más no accesible. Está relacionada con sistemas altamente escalables
FIA.4 Escalabilidad	Habilidad de atender consistentemente las solicitudes a pesar de las variaciones del volumen de la demanda.
FIA.5 Tolerancia a Fallas	Capacidad del producto de software para mantener un nivel de rendimiento específico en caso de errores en el software o de infracciones sobre sus interfaces.
FIA.6 Madurez	Capacidad del producto de software para evitar fallas como resultado de errores en el mismo.

Tabla 5. Características del Modelo correspondiente a la categoría Fiabilidad.

Características	Descripción
USA.1 Documentación	Este concepto está relacionado con la existencia de un documento WSDL donde se expone la funcionalidad, la accesibilidad y la comunicación del WS, entre otros.

Tabla 6. Características del Modelo correspondiente a la categoría Usabilidad.

Características	Descripción
MAB.1 Capacidad de análisis	Capacidad del producto de software para ser diagnosticado por deficiencias o fallas en el software.
MAB.2 Capacidad de cambio	Capacidad del producto de software para implementar una modificación específica de una manera más sencilla.
MAB.3 Estabilidad	Capacidad del producto de software para evitar efectos inesperados después de modificaciones en el software.

Tabla 7. Características del Modelo correspondiente a la categoría Mantenibilidad.

Características	Descripción
POR.1 Adaptabilidad	Capacidad del producto de software para ser adaptado a diferentes ambientes especificados sin aplicar acciones u otros medios que no sean los provistos para este propósito en el software considerado.
POR.2 Co-existencia	Capacidad del producto de software para co-existir con otro software independiente en un ambiente común compartiendo recursos comunes.
POR.3 Capacidad de reemplazo	Capacidad del producto de software para ser usado en lugar de otro producto de software especificado para el mismo propósito en un mismo ambiente. Por ejemplo, la capacidad para el reemplazo de una nueva versión de un producto es importante para el usuario, cuando ésta se actualiza.

Tabla 8. Características del Modelo correspondiente a la categoría Portabilidad.

Característica	Métrica	Pregunta	Tipo de escala
1.1. Precisión	Resultados o efectos correctos o convenientes	Tasa de respuestas correctas.	1
	Ejecución de las actividades	Tasa de actividades que se ejecutan completamente.	2
1.2. Transacción	Correspondencia del estado inicial	¿Cuándo ocurre un error durante una transacción, el estado inicial se mantiene consistentemente?.	4
	Correspondencia del estado final	¿Cuándo una transacción es exitosa, el estado final perdura en el tiempo; es decir; es persistente?.	4
	Cantidad de transacciones exitosas	Tasa de transacciones son completadas exitosamente.	5
1.3.Seguridad	Seguridad de la comunicación	¿El WS utiliza el protocolo SSL (Secure Sockets Layer) para transmitir los datos?.	3
		¿El WS posee algún certificado criptográfico SSL proporcionado por una Autoridad Certificada?.	3
		¿Existe algún mecanismo que permite proteger el certificado SSL?.	3
		¿El WS genera alguna copia de la información confidencial que se transmite?.	3
	Seguridad de la aplicación	¿El WS utiliza Apache como servidor Web?.	3
		¿El WS fue desarrollado utilizando tecnología JAVA?.	3
		¿El WS se encuentra alojado en un lugar físicamente seguro?.	3
	Protección del código fuente del WS	¿El WS posee algún mecanismo que permite proteger su propio código fuente de usuarios no autorizados?.	3
Control del acceso no autorizado	¿El WS posee algún mecanismo para controlar el acceso no autorizado?.	3	
Autenticación	¿El WS posee algún mecanismo de autenticación de usuarios?.	3	
1.4. Interoperabilidad	Existencia de funcionalidades que pertenecen a otro sistema	¿Existen funcionalidades utilizadas por el WS, que pertenecen a otro sistema? .	3
	Existencia de funcionalidades pertenecientes al WS que son utilizadas por otros sistemas	¿Existen funcionalidades utilizadas por otros sistemas, que pertenecen al WS? .	3
	Tasa de funcionalidades utilizadas por el WS, que pertenecen a otro sistema	Tasa de funcionalidades que pertenecen a otros sistemas.	6
	Intercambio de datos	¿Existe intercambio de datos con otros sistemas?.	3
2.1. Estandarización	Ajuste a estándares, convenciones o regulaciones.	¿EL WS se ajusta a los estándares XML, SOAP, UDDI y WSDL?.	3
		¿El WS se ajusta a convenciones o regulaciones existentes?.	3

2.2. Comportamiento del tiempo (Tiempo de respuesta)	Suministro de respuestas y tiempos de procesamiento apropiados en tiempo de ejecución	Tasa del tiempo de respuesta obtenido (real) comparado con el tiempo esperado (ideal).	7
2.3. Utilización de recursos	Velocidad de transmisión de datos	¿Cuál es la velocidad de transmisión del WS?.	8
2.4. Latencia	Medir el tiempo transcurrido entre enviar una solicitud y recibir una respuesta	¿Cuánto tiempo promedio transcurre entre enviar una solicitud y recibir respuesta en un tiempo determinado?.	9
2.5. Troughput	Medir el número de peticiones atendidas por un WS en un período de tiempo determinado	Tasa de Throughput.	10
3.1. Disponibilidad	Capacidad del WS de ser encontrado fácilmente por un sistema	¿El WS puede ser encontrado fácilmente por un sistema?.	3
3.2. Tiempo de reparo (TTR)	Medir el tiempo que toma reparar un servicio que ha fallado	¿Cuánto tiempo toma reparar un servicio que ha fallado?.	11
3.3. Accesibilidad	Capacidad del WS para atender solicitudes	¿El WS es capaz de atender todas las solicitudes que se le pide?.	3
3.4. Escalabilidad	Atención consistente de solicitudes a pesar de las variaciones de la demanda	Tasa de capacidad para atender solicitudes.	12
		Tasa de crecimiento anual.	13
3.5. Tolerancia a fallas	Funcionamiento del WS en caso de errores en el software	¿El WS posee algún mecanismo que permita su funcionamiento aunque existan errores en el software?.	3
3.6. Madurez	Número de fallas como resultado de errores en el WS	¿Cuántas fallas ocurren en el WS en una semana?.	14
	Tiempo promedio entre fallas	Tiempo promedio entre fallas.	14
	Capacidad para resolver fallas	Tasa de fallas ocurridas que son resueltas.	16
4.1. Documentación	Existencia del documento WSDL	¿Existe un documento WSDL donde se describe la funcionalidad del WS?.	3
		¿Existe un documento WSDL donde se describe la accesibilidad del WS?.	3
		¿Existe un documento WSDL donde se describe el mecanismo de comunicación del WS?.	3
5.1. Capacidad de análisis	Facilidad para ser diagnosticado	Grado de facilidad al diagnosticar el software.	17
	Auto explicación del código	¿El código es autoexplicativo?.	3
	Indentación del código	¿El código del WS está indentado correctamente?.	3
	Documentación del código	¿El código del WS está documentado correctamente?.	3
5.2. Capacidad de cambio	Parametrización	¿Es utilizado el pase de parámetros?.	3
	Independencia funcional de los módulos	¿El WS está distribuido en módulos diferentes?.	4
	Diferenciación de las capas: presentación, lógica y datos	¿La capa lógica, presentación y datos son diferenciables entre sí?.	4
	Acoplamiento	¿Un módulo accede al módulo subordinado por él, por medio de una lista convencional de argumentos?.	3
		¿Un módulo accede al módulo subordinado por él, pasando una porción de la estructura de datos?.	3
		¿Un módulo accede al módulo subordinado por él, pasando una variable que controla las decisiones en el segundo?.	3

		¿Los módulos están atados a un entorno externo al software?.	3
		¿Varios módulos hacen referencia a un área global de datos?.	3
		¿Un módulo hace uso de datos o de información de control mantenidos dentro de los límites de otro módulo?.	3
	Cohesión	¿Cada módulo del WS realiza un conjunto de tareas poco relacionadas las unas con las otras?.	3
		¿Cada módulo del WS realiza tareas relacionadas lógicamente?.	3
		¿Cada módulo del WS contiene tareas relacionadas por el hecho de que todas deben hacerse en el mismo intervalo de tiempo?.	3
		¿Cada módulo del WS presenta los elementos de procesamiento relacionados entre sí y deben ejecutarse en un orden específico?.	3
¿Cada módulo del WS presenta a todos los elementos de procesamiento concentrados en un área de la estructura de datos?.	3		
5.3. Estabilidad	Cantidad de variables modificadas con respecto al total de variables en un módulo	Cantidad de variables modificadas con respecto al total de variables en un módulo.	
	Cantidad de variables globales con relación a los módulos que las utilizan	Cantidad de variables globales con relación a los módulos que las utilizan.	18
6.1. Adaptabilidad	Descripción de datos independiente de la plataforma	¿Los datos pueden ser descritos independientemente de la plataforma?.	3
	Plataformas de software donde puede ser operado	Cantidad de sistemas operativos (SO) en los que el WS puede funcionar correctamente.	
6.2. Coexistencia	Co-existencia con otros sistemas	¿Existen otros sistemas compartiendo los mismos recursos?.	3
	Tasa de satisfacción con los productos que co-existen	Tasa de satisfacción respecto a los productos que co-existen en el ambiente.	20
	Integración con otros productos	¿El WS es capaz de interactuar con otros sistemas independientemente de la plataforma?.	3
6.3. Capacidad de reemplazo	Capacidad para reemplazar a otro sistema con el mismo propósito	¿Puede el WS reemplazar a otro sistema?.	21
	Capacidad para utilizar los mismos datos al reemplazar a otro sistema	¿Pueden ser utilizados los mismos datos de un sistema al sustituirlo?.	21

Tabla 9. Métricas para valorar la calidad de los WS.

La Tabla 10 muestra los rangos utilizados en las diferentes escalas.

Escala	Rango
1	Respuestas correctas / Respuestas procesadas
2	Actividades ejecutadas completamente / Actividades totales
3	5 = Si 1 = No
4	5= Siempre, 4= Casi siempre, 3= En ocasiones, 2= Muy poco, 1= Nunca
5	Cantidad de transacciones exitosas / Cantidad de transacciones totales.
6	Funcionalidades que pertenecen a otros sistemas / Funcionalidades del WS
7	Tiempo de respuesta real / Tiempo de respuesta requerido
8	Cant. de datos / 60 seg
9	5 = menos de 15 seg, 4 = entre 16 seg y 30 seg, 3 = entre 31 seg y 1 min, 2 = entre 1 min y 5 min, 1 = más de 5 min
10	Cantidad de peticiones atendidas / número de peticiones
11	5 = menos de 1 min, 4 = entre 1 min y 10 min, 3 = entre 10 min y 20 min, 2 = entre 20 min y 1 hora, 1 = más de 1 hora
12	Cantidad de solicitudes que atiende actualmente / cantidad de solicitudes que es capaz de atender
13	Cantidad de solicitudes que es capaz de atender actualmente / cantidad de solicitudes que es capaz de atender en un año

14	4=Ninguna, 3=Muy pocas, 2=Algunas, 1=Muchas
15	5=Semestral o más, 4=Trimestral, 3= Mensual, 2= Semanal, 1= Diario
16	Cantidad de fallas resueltas / Cantidad de fallas ocurridas
17	5=Muy fácil, 4=Fácil, 3=Promedio, 2=Difícil, 1= Muy difícil
18	5 = Muy baja, 4= Baja, 3 = Media, 2 = Alta, 1 = Muy alta
19	5 = 5 SO, 4 = 4 SO, 3 = 3 SO, 2 = 2 SO, 1 = 1 SO
20	Nº de productos requeridos con los que co-existe / Nº de productos requeridos
21	5 =Totalmente, 4 = Casi todo, 3 = Medianamente, 2 = Poco, 1 =Ninguno

Tabla 10. Tipos de escala para las métricas.

5. EVALUACIÓN DEL MODELO

Para llevar a cabo la evaluación del Modelo propuesto, se aplicó el Método Análisis de Características [11] en dos Estudios de Caso; los pasos seguidos [11] fueron los siguientes: a) identificación de los criterios de la escala de evaluación, b) aplicación del Modelo de Especificación de calidad para WS a dos proyectos, c) evaluación de los criterios del Modelo de Especificación de calidad para WS, y d) análisis de la valoración y presentación de resultados.

5.1 Identificación de los criterios de la escala de evaluación

Se tomó como referencia los pasos que señalan Kitchenham y Jones [11]; por ello fue necesario establecer un conjunto de aspectos a ser verificados que van desde lo más general, el modelo propuesto, a lo más específico, que son las métricas. Según Kitchenham y Jones existen dos tipos de criterios [11]:

- **Simple.** Se utilizan cuando el criterio se encuentra presente o ausente en el contexto de evaluación.
- **Compuestos.** Son utilizados cuando la existencia o conformidad de un criterio puede ser medida con una escala ordinal.

Para efectos de esta evaluación se utilizaron sólo los criterios o aspectos simples para obligar al evaluador a establecer un criterio claro de aceptación (ver Tabla 11).

Escala
1: significa que el Modelo o métrica tiene el criterio establecido.
0: significa que el Modelo o métrica no tiene el criterio establecido.

Tabla 11. Escala utilizada para evaluar los criterios generales y específicos.

Los criterios más generales evalúan el Modelo en un nivel macro, su definición conceptual se presenta en la Tabla 12.

Criterios General	Descripción
Pertinencia del Modelo de Calidad	Se refiere a si el Modelo de Calidad propuesto es pertinente o no dentro del proceso de especificación de la calidad del software.
Complejidad de las categorías involucradas	Se refiere a si las Categorías: Funcionalidad, Eficiencia, Fiabilidad, Usabilidad, Mantenibilidad, y Portabilidad; dan cobertura total a los aspectos de calidad de los WS.
Adecuación al contexto	Se refiere a si la especificación de la calidad Modelo propuesto es adecuada en el contexto de la evaluación.
Precisión del nivel de calidad especificado por el Modelo propuesto	Se refiere a si la calidad especificada por el Modelo propuesto en el proyecto piloto fue precisa.

Tabla 12. Criterios Generales a evaluar para el Modelo de Especificación de Calidad de WS.

Una vez especificadas los criterios generales fue necesario establecer un conjunto de criterios que permitan evaluar las métricas del Modelo. Los criterios específicos a considerar se presentan en la Tabla 13. Culminada la identificación de los criterios que permitirán la evaluación, se ha establecido el criterio de aceptación a utilizar. Se tomó como nivel de aceptación de los criterios un valor del 75%. Este porcentaje de aceptación fue tomado por consenso de los equipos evaluadores y los autores, considerando que es una práctica común en la mayoría de los modelos de calidad.

Criterios Específicos	Descripción
Pertinencia de la métrica	Se refiere a si una métrica es adecuada para medir la existencia o no de la característica donde se encuentra.
Factibilidad de la métrica	Se refiere a si es factible medir la característica propuesta en la métrica dentro del contexto de evaluación.
Nivel de profundidad	Se refiere a si la métrica a verificar tiene el nivel de profundidad adecuado para que el resultado sea relevante.
Escala de la métrica	Se refiere a si la escala propuesta es adecuada para medir la métrica.

Tabla 13. Criterios Específicos a evaluar para el Modelo de Especificación de Calidad de WS.

5.2 Aplicación del Modelo de Especificación de calidad para WS a dos proyectos

Luego se aplicó el Modelo a dos estudios de caso, para ello se utilizaron dos WS. Según Kitchenham y Jones [11], para aplicar este método de evaluación, de debe configurar un equipo evaluador. Para este caso se configuraron dos grupos de cuatro personas, dos de ellas investigadores de LISI y dos desarrolladores de cada WS para cada uno de los estudios de caso. La selección del WS a utilizar para la aplicación del Modelo de Especificación de la Calidad propuesto, dependió de la disponibilidad de acceso a los mismos. En este sentido, se logró el acceso a dos WS, el primero es un WS tipo WSA y el segundo WS es del tipo WSC. El grupo evaluador especificó la calidad de cada WS utilizando el Modelo propuesto. Para ello se aplicó el mismo algoritmo que se propone para MOSCA [17]. Después, haciendo uso de los criterios generales y específicos propuestos anteriormente, se determinó la calidad del Modelo.

5.3 Evaluación de los criterios de Modelo de Especificación de calidad para WS

Las figuras 3 y 4 muestran los resultados de la evaluación de las **Criterios Generales** y de las **Criterios Específicos** del Modelo.

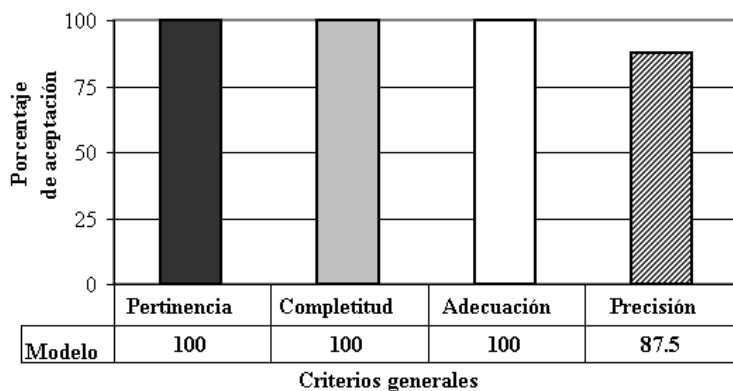


Figura 3 .Resultados criterios generales del modelo.

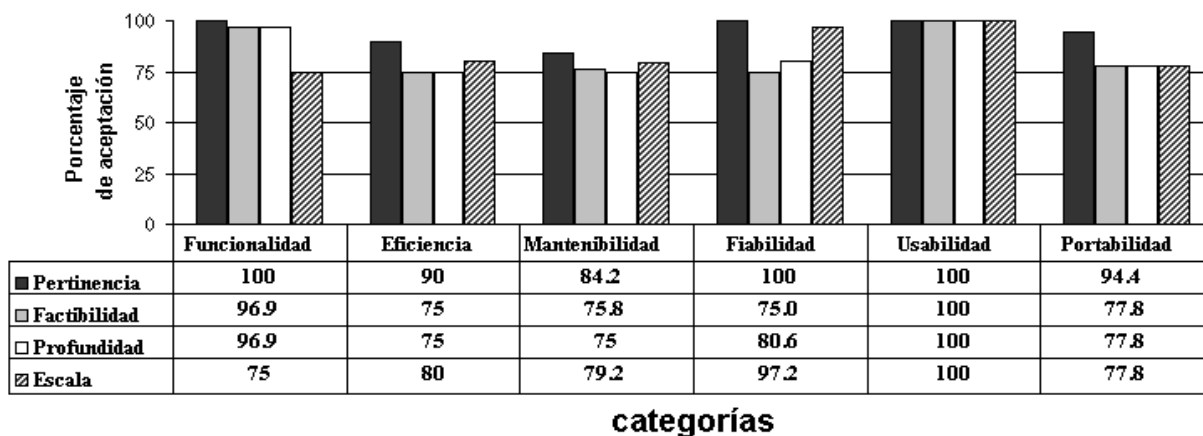


Figura 4. Resultados criterios específicos del modelo por categorías.

5.4 Análisis de la valoración y presentación de resultados

De la aplicación del método de evaluación se obtienen dos tipos de resultados. El primer tipo se refiere al Modelo de Calidad propuesto; es decir, los valores que se obtienen por los criterios. El segundo tipo se refiere a la estimación de calidad que se obtiene sobre los dos WS evaluados.

Sobre la evaluación del modelo se obtiene que todos los criterios generales alcanzaron el máximo porcentaje en la evaluación, a excepción del criterio precisión que alcanzó un 87,5% de satisfacción (Figura 3), considerando que el Modelo de Calidad es **Pertinente** dentro del proceso de especificación de la Calidad del Software. Además, estuvieron de acuerdo en que las categorías Funcionalidad, Eficiencia, Fiabilidad, Usabilidad, Mantenibilidad, y Portabilidad, abarcan **Completamente** los aspectos de calidad de los WS. Siendo parte del contexto de evaluación, los evaluadores consideraron **Adecuada** la calidad del Modelo en dicho contexto.

.En cuanto a la evaluación de cada categoría (Figura 4) se obtuvo que de igual forma cada una por separado es **Pertinente** en cuanto al proceso de estimación de la calidad, es **Completa** en cuanto a su especificación, es **Adecuada** en el contexto de evaluación y **Precisa** en el resultado alcanzado, excepto en la categoría Eficiencia donde la precisión alcanzó un 83% de satisfacción. Esto se debe a que en ciertas métricas de dicha categoría la escala de evaluación es muy subjetiva. Las categorías Funcionalidad, Fiabilidad y Usabilidad fueron consideradas **Pertinentes** en el contexto de evaluación. Le sigue en porcentaje de pertinencia de las métricas la Portabilidad con un 94,43%, luego la Eficiencia resultó obtener un 90% y, por último, la Mantenibilidad con un 84,2% de pertinencia de sus métricas. De acuerdo al criterio de aceptación las categorías resultaron obtener en promedio más del 75% por lo que se les considera aceptables. En el caso de la **Factibilidad** los evaluadores consideraron que en la categoría Usabilidad, el 100% de las métricas eran factibles de evaluar en el contexto en que se aplicaron; seguidamente se encuentra la categoría Funcionalidad con un 96,9% y por último las categorías Portabilidad, Mantenibilidad y Eficiencia y Fiabilidad con un porcentaje de satisfacción mayor o igual al 75%. En el caso de la Profundidad se puede observar en la Figura 5 que para los evaluadores en promedio el 84,21% de las métricas de cada categoría presentaban la **Profundidad** adecuada. Los evaluadores consideraron que el 100% de las métricas de la Usabilidad posee la **Escala** apropiada. Seguidamente las escalas de las métricas para especificar la Funcionalidad, Mantenibilidad, Fiabilidad y Portabilidad, obtuvieron un promedio mayor o igual al 75%. **Puesto que el promedio para cada característica supera ampliamente el 75% establecido en el criterio de aceptación se considera entonces que el modelo es pertinente, completo, adecuado y preciso, respecto a la evaluación llevada a cabo a los dos WS.**

Sobre la calidad de los WS, la estimación de los evaluadores arrojó un nivel de calidad **Intermedio** para el primer WS y **Avanzado** para el segundo, coincidieron en que el nivel de calidad especificado fue preciso, aunque en la categoría Eficiencia no proporcionó el nivel de precisión esperado por los evaluadores.

6. CONCLUSIONES

En esta investigación se propuso un Modelo de Especificación de la Calidad para Web Services, basado en características, con un total de 46 métricas. El principal aporte de este modelo es que parte de una base ya probada (MOSCA), el cual a su vez se inspira en estándares de calidad y propone métricas, para luego incorporar los aspectos propios de la calidad de WS y QoS, donde éste último es también una fuente de requerimientos importante íntimamente relacionados al mundo de la Ingeniería Web y las aplicaciones en Internet. El desarrollo de este modelo confirma que la calidad no es inflexible sino contextual. En este modelo se evaluaron las seis categorías: Funcionalidad, Eficiencia, Mantenibilidad, Fiabilidad, Usabilidad y Portabilidad. El estudio realizado a través de los dos WS arroja que al estimar la calidad de un WS se debe enfocar la misma en las categorías Eficiencia y Fiabilidad; considerando igualmente el aspecto de la Seguridad. No así la Usabilidad, la cual pierde relevancia para los Web Services por ser éstos una capa middleware con la que el usuario no actúa directamente. La evaluación del Modelo a través de dos WS, mostró que es pertinente a la calidad de los WS, completo en cuanto a su especificación, adecuado en el contexto de evaluación y preciso en el resultado alcanzado.

7. AGRADECIMIENTOS

Esta investigación fue financiada por el Fondo Nacional de Ciencia, Tecnología e Innovación (FONACIT) de la República Bolivariana de Venezuela, a través de los proyectos S1-2000000437 y S1-2001000792

8. REFERENCIAS

- [1] Basili, V. y Weiss, D. A Methodology For Collecting Valid Software Engineering Data, *IEEE Trans. Software Eng. SE*, (1984) 10(6), pp. 728-738.
- [2] Callaos, N. y Callaos, B., Designing with a systemic total quality, *Proceedings of the International Conference on Information Systems Analysis and Synthesis*, ISAS'96, (1996), pp. 15-23.
- [3] Canta, C., Rufino, E., Restuccia, R. y Ruiz, F. *Web services*, Trabajo de investigación, Universidad Católica de Salta, Salta, Argentina, (2003).
- [4] Dromey, G., A Model for Software Product Quality, *IEEE Transactions on Software Engineering* 2(2), February, (1995), pp. 146-162.
- [5] Gillies, A. *Software Quality: Theory and Management*, Thomson, 2nd edition, (1997).
- [6] Hangjung, Z. Designing Intra and Inter Organizational Business Processes with Web Services, Tampa, USA: *AMCIS. Ninth Americas Conference on Information Systems*, (2003).
- [7] Hansen, G. Quality of Services, (1997), consultado el 05/05/2005 en: <http://www.objs.com/survey/QoS.htm>
- [8] Humphrey, W. *Managing the Software Process*, Addison-Wesley Pub. Co., London, UK, (1998), pp.7.
- [9] ISO/IEC 9126 1.2. Information Technology-Software Product Quality, Part 1, Quality Model, *ISO/IEC JTC1/SC7/WG6*, (1998).
- [10] ISO/IEC TR 15504-2. Information Technology-Software Process Assessment, Part 2, A Reference Model Processes and Process Capability, Canada, *ISO/IEC JTC 1/SC 7*, (1998).
- [11] Kitchenham, B., Jones, L Evaluation Software Engineering Methods and Tools. Part 6: Identifying and Scoring Features, *ACM Software Engineering Notes*, (1997), Vol. 22, No. 2.
- [12] Losavio, F., Ortega, D. y Perez, M. Towards a Standard EAI Quality terminology, *Proceedings of XXIII International Conference of The Chilean Computer Science Society*, (2003) pp. 119-129.
- [13] Maamar, Z. y Alkhatib, G. Integration of Web Services for Establishing Virtual Enterprise: An Agent-Based Perspectiva, Tampa, USA: *AMCIS. Ninth Americas Conference on Information Systems*, (2003).
- [14] Mani, A. y Nagarajan, A. Understanding Quality of service for Web Services, (2002), consultado el 05/05/2005 en: <http://www-106.ibm.com/developerworks/webservices/library/ws-quality.html>
- [15] Microsoft business solutions. E-business - *White paper*, (2003), consultado el 15/01/2004 en: <http://www.dei.isep.ipp.pt/~qtdei/qtdei/MBSPT%20%20E-biz%20white%20paper.pdf>
- [16] Murray, M. An Inicial Investigation of Web Services in Healthcare, Kennesaw State University, Tampa, USA: *AMCIS. Ninth Americas Conference on Information System*, (2003).
- [17] Mendoza, L., Pérez, A., Griman, A. y Rojas, T. Algoritmo para la Evaluación de la Calidad Sistémica del Software, *2das. Jornadas Iberoamericanas de Ingeniería del Software e Ingeniería del Conocimiento, IIISIC 2002*, (2002) Brasil.
- [18] Mendoza, L., Pérez, M. y Grimán, A. Prototipo de Modelo Sistémico de Calidad (MOSCA) del Software. *Computación y Sistemas Revista Iberoamericana de Computación*, (2005), Vol. VIII, No. 3, Mexico.
- [19] Newcomer, E. *Understanding Web Services. XML, SOAP, and UDDI*, Boston, USA: Addison Wesley, (2002).
- [20] Ortega, M., Pérez, M. y Rojas, T. Construction of a Systemic Quality Model for evaluating a Software Product. *Software Quality Journal*, (Julio 2003), 11(3), pp. 219-242.
- [21] Perez, M., Rojas, T., Ortega, M., y Caldera, A. Toward Systemic Quality: Case study, 4Th Squad Meeting, Venezuela (1999).
- [22] Pérez, L. M., Rojas, T., Mendoza, L. y Griman, A. Systemic Quality Model for System Development Process: Case Study, *Proceedings of AMCIS 2001*, (2001).
- [23] Pérez, M., Mendoza, L. y Grimán, A. Hacia una Certificación de la Calidad Sistémica en los Sistemas de Software, *IV Congreso de Investigación y Creación Intelectual de la UNIMET*, (Mayo 2004), Caracas, Venezuela.
- [24] Rincón, G., Mendoza, L. y Pérez, M. Guía para la Adaptación de un Modelo Genérico de Software, *IV Jornadas Iberoamericanas en Ingeniería del Software e Ingeniería del Conocimiento*, (Noviembre 2004), Madrid, España.
- [25] Systinet Corp. Introduction to Web Services Architecture, consultado el 16/01/2004 en: www.systinet.com.
- [26] Voas, J. Software Quality's Eight Greatest Myths, *IEEE Software*, 16(5), September/October, 1999, pp. 740-745.