

La eficacia de la evaluación externa de un producto de software orientado a la web basado en lógica difusa

Effectiveness evaluation of a software product oriented web based on fuzzy logic

Msc. Erlington Salcedo^{1*}, Esp. Celio Gil Aros^{2*}

¹ Msc. en Tecnologías de Información. Profesor de Ingeniería de Sistemas. Fundación Universitaria Los Libertadores, Bogotá, Colombia. Grupo GRIDNTIC. Proyecto: Sistema difuso para la evaluación de la calidad externa de software orientado a la web. Investigador Principal. Director del grupo de Investigación. *esalcedob@libertadores.edu.co.

² Esp. en Administración de Empresas. Profesor de Ingeniería de Sistemas. Bogotá, Colombia. Fundación Universitaria Los Libertadores. Grupo GRIDNTIC. Proyecto: Sistema difuso para la evaluación de la calidad externa de software orientado a la web. Co-investigador. *cegil0056@yahoo.es.

Fecha de recepción del artículo: 8/9/2011 Fecha de aceptación del artículo: 2/11/2011

Resumen

En este artículo se presenta un estudio detallado que permite calificar la calidad de un producto de software aplicando un sistema desarrollado con lógica difusa, el cual permite evaluar las respuestas que dicho sistema genera a partir de las consideraciones que se deben tener en cuenta para la definición de calidad, dados los parámetros que los usuarios finales buscan en estos sistemas.

La metodología aplicada es cuasi-experimental, dada a partir de encuestas que ayudan a definir las características y niveles de calidad que cada usuario tiene, para luego plasmar dichos parámetros en un producto de software, que calificará tres sitios web bajo dichas características.

A partir de esto se encontró que los resultados obtenidos por el sistema, frente a los definidos por el usuario son parecidos y por ende no tienen diferencias significativas entre sí, llevando a poder asegurar que el software evalúa la calidad de sistemas web de forma eficaz.

Palabras clave

Lógica Difusa, Métricas de Software, Variables Lingüísticas, Sistema de Inferencia, Calidad de Software.

Abstract

This paper presents a detailed study that can be qualified as a software product developed by applying a fuzzy logic system, which allows evaluating the responses that the system generated from the considerations that must be taken into account in the definition of quality, given the parameters that end users look for in these systems.

The methodology is quasi-experimental, based on surveys given to help define the characteristics and quality standards that each user has and then translate these parameters into a software product, which will qualify three websites on these characteristics.

From this it was found that the results obtained by the system, compared to those defined by the user are similar and therefore have no significant differences among them, being able to ensure that the software evaluates the quality of web systems effectively.

Keywords

Fuzzy Logic, Software Metrics, Linguistic Variable, Inference System, Quality Software.

Introducción

Según recientes estudios, los productos de software deben ser desarrollados en un menor tiempo, con la menor cantidad de recursos y cada vez con mayores requerimientos por parte de los usuarios [2], conllevando todo esto a que la ingeniería de software diseñe nuevos métodos, procesos y herramientas para la construcción de sistemas con una mayor confiabilidad, estabilidad, amigabilidad, parametrización, eficiencia, seguridad y calidad, entre otros, según definen varios autores [5]. De esta forma se plantean modelos de calidad, como un proceso de mejora continua necesaria [6]. Así, mismo se necesita una guía cuantitativa que proporcione a los desarrolladores e ingenieros una forma de explorar el diseño y la construcción de software, formulándose medidas (métricas) e indicadores [7], las cuales evalúan con mayor precisión las diferentes potencialidades del sistema y necesidades del usuario. Todas estas actividades persiguen una medida estándar, donde existen datos e información difíciles de cuantificar y donde la incertidumbre es una variable genérica y subjetivista, mostrando una realidad confusa dado el ruido generado por este tipo de situaciones.

Si bien es cierto el conjunto de métricas internas de software son dadas a partir de las líneas de código fuente y de la toma de tiempos en los diferentes procesos de respuesta de la aplicación y demás medidas que se pueden generar de forma directa por otros ingenieros o personas, las cuales pueden ser más o menos precisas [8], al hablar de métricas externas, el cual es el motivo de esta investigación, el panorama no es muy alentador, dado que una medición externa ya no está definida por una ecuación matemática planteada a partir de la toma de tiempos o de la visualización de los procesos del sistema, sino a partir de lo que “*Siente y Percibe*” el usuario final cuando interactúa con un aplicativo y es aquí donde la lógica difusa entra a formar parte de la elaboración de un sistema de software que busca captar dichas percepciones y a partir de ellas generar una estandarización de esas métricas y la obtención de un factor de calidad total del sistema,

el cual aporte a la calificación del sitio web que busca lograr la calidad a nivel externo.

Dadas las anteriores consideraciones se formula la pregunta: ¿Cuál es el grado de eficacia de la evaluación de la calidad externa de un producto de software orientado a la Web por medio de un sistema basado en Lógica Difusa con respecto a la evaluación realizada por un experto?, planteando para su desarrollo, la definición de un modelo que permita evaluar los diferentes factores de calidad de software orientado a la web a través de modelamiento matemático, donde la lógica difusa permite crear un proceso sistemático para dar solución al problema.

Para efectos de esta investigación solo se va a trabajar la calidad externa, ya que la calidad interna se ha estudiado con mayor profundidad y se ha logrado, no solo definirla, sino trabajarla a partir de diferentes métricas. Por su lado la calidad externa, ha sido poco estudiada y su medición se ha realizado a través de métricas donde los procesos se centran en lo que el usuario está captando mientras interactúa con el sistema, por lo cual son muy difíciles de medir y no hay una forma específica de trabajo que permita estandarizar medidas que generen una calificación de su calidad.

Marco teórico

La calidad de un producto de software es el conjunto de cualidades que lo caracterizan y que determinan su utilidad y existencia frente a estándares, planteando un adecuado equilibrio en factores como eficiencia, confiabilidad, seguridad e integridad, entre otros, la cual puede definirse bajo dos aspectos [4]:

- **Calidad Interna:** Medible a partir de las características intrínsecas, como el código fuente.
- **Calidad Externa:** Medible en el comportamiento del producto, evaluada por el usuario final.

La calidad externa es subjetiva en función de la persona que evalúa el producto y del cumplimiento

que el software tenga de los requerimientos funcionales, la forma en que lo hacen, la facilidad de los procesos y demás variables que el cliente busca al momento de utilizar e interactuar con un programa. De esta manera la calidad externa es sinónimo de amigabilidad, facilidad de manejo, escalabilidad, portabilidad, funcionalidad, seguridad, eficiencia, confiabilidad, integridad, como lo definen algunos autores [3].

Al tener este problema definido y una necesidad clara de medir la calidad externa, se compararon modelos tales como el de Mc Call, Boehm, ISO 9126 e ISO 14598 [9], a partir de estos se obtuvo el modelo de calidad que permitió analizar diferentes métodos y conceptos, encontrando en la lógica difusa una ciencia de origen preceptivo del razonamiento, concebida por Lofti A. Zaded [10] la cual tiene como objetivo el análisis multivariado de modelos no lineales, con altos niveles de incertidumbre, a partir de información de entrada vaga, ambigua, imprecisa, con ruido e incompleta, siendo la lógica Difusa, un método de fácil comprensión y adaptabilidad a casos particulares, permitiendo no requerir de algoritmos muy sofisticados y prevaleciendo más el sentido común que la misma matemática para la solución de dichos problemas. Frente a otras herramientas, la lógica Difusa presenta un campo de acción estándar muy estudiado y que ha dado muy buenos resultados tal como se demostró en otras investigaciones [12] y por ende se eligió como la herramienta base de análisis de la presente investigación.

Al evaluar la forma en la cual las personas generan una medida sobre alguna característica, se observa que no se presentan verdades o falsedades absolutas como en la lógica computacional binaria, sino que se define un rango de datos posibles, lo cual imbrica con la forma de trabajo de la lógica difusa [11], capturando y procesando de esta manera, datos inciertos que el evaluador genera.

- **El bloque difusor / congresor:** Es donde a cada variable de entrada se le asigna un grado de pertenencia, relacionando cada una de las

variables con conjuntos difusos de entrada, que representan la reglamentación que delimitará el sistema [13]. De la misma manera el bloque congresor realiza el proceso inverso, donde a partir del conjunto difuso de salida generado por la máquina de inferencia, se generan procesos matemáticos de desdifusión, dando como resultado un valor concreto en términos verbales. De esta manera la *función de pertenencia* de un conjunto nos indica el grado en que cada elemento de un universo está involucrado con dicho conjunto.

- **Motor de Inferencia o máquina de inferencia:** Es la encargada de relacionar los conjuntos de entrada y salida. Todo lo que entra a este bloque son conjuntos difusos o membrecías [13] y la salida también se presentan como conjuntos difusos asociados a una variable, los cuales están dentro de las reglas difusas que definen el sistema. Las reglas deben ser completas en tanto la combinación de los valores de entrada para obtener un valor apropiado de salida, adicional a esto, deben ser consistentes porque no hay contradicciones entre ellas, o sea que no pueden existir dos reglas con iguales antecedentes y distintos consecuentes y por último, son continuas al definir funciones de pertenencia en los consecuentes con intersección no nula.

Metodología

El desarrollo del proyecto implicó ejecutar una serie de etapas, cada una de las cuales requiere la aplicación de una metodología distinta de trabajo. Para comenzar se aplicó la “Metodología de investigación preliminar”, la cual tuvo por objeto la definición del marco teórico, los procesos de recolección de datos y la formulación de la hipótesis, que para este caso es la no existencia de diferencias significativas en los resultados obtenidos en la calificación de un sistema Web por medio de un software elaborado con lógica difusa frente a la evaluación realizada por los usuarios finales.

Posteriormente se definió la metodología para el análisis y la selección de productos de software,

métricas e indicadores, en la cual se desarrollaron encuestas a 200 personas, preguntando cuales son los parámetros de calidad que ellos plantean y los sitios web que más utilizan, con el fin de recolectar ésta información, clasificarla y analizarla para poder involucrarla en el software. A partir de éstos insumos se definieron las variables que se van a tener en cuenta dentro del desarrollo de la investigación, las cuales son:

Evaluación de la calidad externa de un producto de software orientado a la Web (ECE), la cual es la variable principal y sobre ella gira toda la investigación y que según los usuarios finales, la plantean en términos de cuatro sub variables a tener en cuenta:

- **Contenido:** Esta variable está enmarcada bajo dos aspectos, el primero es la definición de un *contenido completo*, el cual plantea una medida de la satisfacción que el usuario tiene por el contenido recibido y la segunda es el *contenido actualizado*, la cual permite definir si el contenido que se está presentando es veraz, consistente y reciente como lo define Calero [1].
- **Interfaz gráfica:** Este parámetro se divide en dos ítems, el primero es el *nivel de agrado* que genera el sitio web en el usuario, el cual está planteado por la utilización de colores, gráficas, tamaños y fuentes y el segundo en términos de la *distribución gráfica* de los componentes, evaluando la ubicación y la presentación de las diferentes partes del sistema dentro de la pantalla, dada a partir de los menús, botones, cabeceras, enlaces, zonas de trabajo, contadores, encuestas y demás posibilidades de contenido tal como lo afirma Gall [2].
- **Navegabilidad:** Está dada a partir de la facilidad para acceder a las diferentes partes del sitio, sin necesidad de definir instructivos para su uso o en su defecto la memorización de caminos de enlaces, que permitan llegar a un contenido puntual, comprobando además la ausencia de enlaces rotos, como lo define el estándar W3C.
- **Seguridad:** Definida como la posibilidad que tiene el sistema para poder acceder a contenidos por medio de jerarquías de usuario, accediendo

por medio de una autenticación y la adjudicación de perfiles de usuario. Este parámetro también lo tiene en cuenta Calero [1].

De esta manera la variable que se analizó es la *eficacia de la evaluación del sistema creado con lógica difusa frente a la medición que hace un usuario final sobre el producto de software orientado a la web*, dada como el nivel de acierto frente a pruebas estadísticas variacionales de tendencia central (ANOVA).

Así mismo, como resultado de la encuesta se definió que el sitio web más utilizado es Google, con un puntaje de 111 votos de una muestra de 200, en segundo lugar se tiene a Hotmail y por último a Facebook con 45 y 35 votos respectivamente. De la misma manera se planteó el sitio web de la universidad con el fin de poder generar conclusiones que permitan dar una calificación de calidad al sitio, frente a lo que piensan las personas evaluadas, pese a que obtuvo el 5° lugar.

Una vez concluido este análisis, se procedió a realizar una segunda encuesta en la cual cada persona asigna un valor a las variables definidas anteriormente para cada uno de los sitios Web elegidos. Esta información no solo da pie para caracterizar cada uno de los parámetros del sistema de lógica difusa sino también para generar las respectivas pruebas del software y su tabulación.

En la etapa de ingeniería se buscó un modelo que cumpla con las condiciones tiempo de desarrollo corto, estabilización rápida del modelo, pocos recursos, definición de equipos de trabajo para las fases, etapa de requerimientos fuerte, modelo orientado a investigación, bondades cumplidas plenamente por el desarrollo rápido de aplicaciones (**DRA**) [6], frente a otros modelos como: Cascada Simple, Sashimi, Espiral, Prototipos, entre otros, razones por las cuales se seleccionó en la construcción del software, teniendo como base el ambiente de desarrollo **Matlab**, el cual tiene una infraestructura de Lógica Difusa que permite parametrizar y generar el aplicativo. En esta parte se definieron variables de entrada

(contenido completo, contenido actualizado, nivel de agrado, distribución gráfica, navegabilidad y seguridad). Cada una de las cuales definen un conjunto difuso, que para el caso estudiado es el mismo, $e(x)=\{\text{excelente, bueno, regular, malo}\}$, dado un universo del discurso $[0,10]$, donde 0 es malo extremo y 10 es excelente extremo y donde las funciones de pertenencia para cada una de las variables se pueden observar en las Figuras 1 y 2.

El sistema difuso utiliza operadores básicos de Zadeh parametrizados como sigue:

- Método para AND: Función de elección por Mínimos.
- Método para OR: Función de elección por Máximos.
- Método para IMPLICACIÓN: Función de elección por Mínimos.

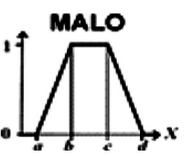
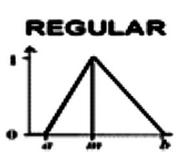
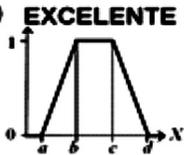
<p>Malo: Está representada por una función trapezoidal con valores $a=-0.36, b=-0.04, c=0.1019$ y $d=0.398$.</p>	<p>Regular: Está representada por una función triangular con valores $a = 0.197, m = 0.4511$ y $b = 0.702$.</p>
<p>Buena: Está representada por una función triangular con valores $a = 0.501, m = 0.689$ y $b = 0.9008$.</p>	<p>Excelente: Está representada por una función trapezoidal con valores $a=0.702, b=0.903, c=1.04$ y $d=1.36$</p>
$E(x) = \begin{cases} 0 & \text{si } (x \leq a) \text{ o } (x \geq d) \\ \frac{(x-a)}{(b-a)} & \text{si } x \in (a,b) \\ 1 & \text{si } x \in (b,c) \\ 0 & \text{si } x \in (c,d) \end{cases}$ 	$E(x) = \begin{cases} 0 & \text{si } x \leq a \\ \frac{(x-a)}{(m-a)} & \text{si } x \in (a,m) \\ \frac{(b-x)}{(b-m)} & \text{si } x \in (m,b) \\ 0 & \text{si } x \geq b \end{cases}$ 
$E(x) = \begin{cases} 0 & \text{si } x \leq a \\ \frac{(x-a)}{(m-a)} & \text{si } x \in (a,m) \\ \frac{(b-x)}{(b-m)} & \text{si } x \in (m,b) \\ 0 & \text{si } x \geq b \end{cases}$ 	$E(x) = \begin{cases} 0 & \text{si } (x \leq a) \text{ o } (x \geq d) \\ \frac{(x-a)}{(b-a)} & \text{si } x \in (a,b) \\ 1 & \text{si } x \in (b,c) \\ 0 & \text{si } x \in (c,d) \end{cases}$ 

Figura 1. Funciones de pertenencia.

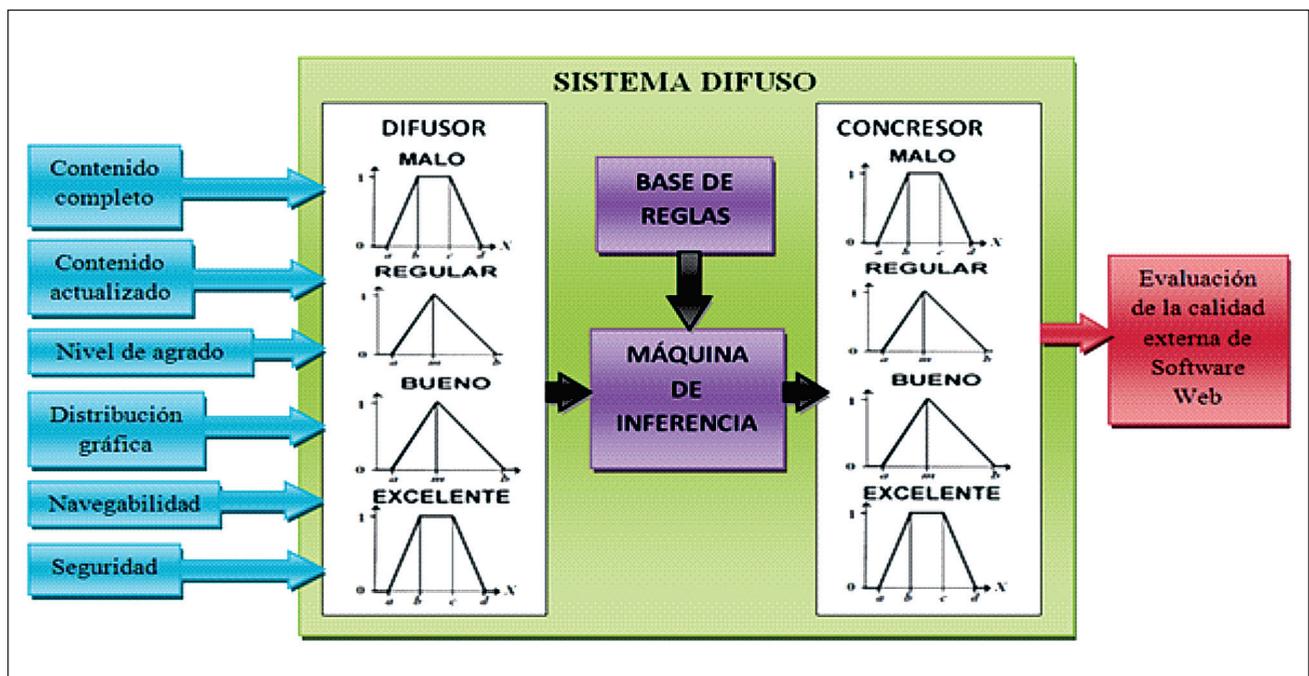


Figura 2. Diseño del sistema de software (Fuente: Autores).

- Método para AGREGACIÓN: Función de elección por Máximos.
- Método para DEFUSIFICACIÓN: Función de elección por centroide.
- Algoritmo de agrupamiento “fuzzy Isodata”.

Bajo las anteriores definiciones el sistema se diseñó de la siguiente manera:

Análisis de resultados

Con base en los datos de las respectivas encuestas, se elaboró un análisis para cada uno de los sitios web junto con los resultados obtenidos por el software.

Análisis para Google

Como $F_{0,05}(3,6) = 4.757$ y de la tabla de Anova tenemos que $F = 2.432$ y $2.432 < 4.757$, por lo tanto aceptamos que los resultados obtenidos por el sistema, frente a los definidos por el usuario son parecidos y por ende no tienen diferencias significativas entre sí. Por otro lado la prueba T para muestras relacionadas sobre un nivel de significancia de 0.05, define que se acepta que el software genera una calificación parecida a la del usuario (Tablas 1 y 2).

Análisis para Facebook

Como $F_{0,05}(4,5) = 5.192$ y de la tabla de Anova tenemos que $F = 8.877$ y $8.877 > 5.192$, por lo tanto no se acepta que los resultados obtenidos por el sistema, frente a los definidos por el usuario sean parecidos y por ende hay diferencias significativa entre sí. Con la prueba T se llega a la conclusión de definir que no hay un buen porcentaje de aceptación del software (Tablas 3 y 4).

Análisis para Hotmail

Como $F_{0,05}(3,6) = 4.757$ y de la tabla de Anova tenemos que $F = 4.09$ y $4.09 < 4.757$, por lo tanto se acepta que los resultados obtenidos por el sistema, frente a los definidos por el usuario son parecidos

y por ende no hay diferencias significativa entre sí, corroborándose con la prueba T (Tablas 5 y 6).

Análisis para la Página de la Fundación Universitaria Los Libertadores

Como $F_{0,05}(5,4) = 6.256$ y de la tabla de Anova tenemos que $F = 4.421$ y $4.421 < 6.256$, por lo tanto se acepta que los resultados obtenidos por el sistema, frente a los definidos por el usuario son parecidos y por ende no hay diferencias significativa entre sí, aceptándose que el software realiza una buena calificación a partir del análisis definido por la prueba T.

Bajo los parámetros anteriores podemos inferir que el sistema de lógica difusa, según las pruebas T, es eficiente y se puede tomar como un sistema confiable que permite evaluar la calidad externa de un sistema Web.

Conclusiones

El desarrollo teórico e investigativo adelantado, permite correlacionar la calidad de software con las métricas y poder utilizar la lógica difusa como mediador entre estos conceptos, llegando a la conclusión de aceptar la hipótesis en la cual no existen diferencias significativas en los resultados obtenidos en la calificación de un sistema Web por medio de un software elaborado con lógica difusa, frente a la evaluación realizada por los usuarios finales. Las pruebas estadísticas Anova y prueba T, permitieron comprobar que el sistema difuso se comporta de manera similar a la tendencia de un ser humano, al generar resultados parecidos en tres de los cuatro casos planteados. Además, se puede afirmar que la lógica difusa es una herramienta aplicable en la toma de decisiones de esta índole [12], tal como lo encontraron varios autores.

También se puede concluir que la lógica difusa es una herramienta ideal para evaluar la calidad externa de sitios web de forma automática, dada la necesidad de estandarizar información y las posibilidades que presenta dicha teoría para su

Tabla 1. Anova y Correlaciones para Google.

ANOVA						Correlaciones de muestras relacionadas				
Sistema2										
	Suma cuad.	gl	Mediacuad.	F	Sig.		N	Correl	Sig.	
Inter-grupos	,100	3	,033	2,432	,163	Par 1	Usuario y Sistema2	10	,279	,434
Intra-grupos	,082	6	,014							
Total	,182	9								

Tabla 2. Prueba de muestras correlacionadas para Google.

Prueba de muestras relacionadas									
Diferencias relacionadas									
		Med	Desviac. Típica	Error típico de la media	95% Inter.confianza para la dif.		t	gl	Sig. (bilateral)
					Inf.	Sup.			
Par 1	Us - Sist2	,1213	,15534	,04912	,01018	,23242	2,469	9	,036

Tabla 3. Anova y Correlaciones para Facebook.

ANOVA						Correlaciones de muestras relacionadas				
Sistema2										
	Suma cuad.	gl	Mediacuad.	F	Sig.		N	Correl	Sig.	
Inter-grupos	,230	4	,057	8,877	,017	Par 1	Usuario y Sistema2	10	,900	,000
Intra-grupos	,032	5	,006							
Total	,262	9								

Tabla 4. Prueba de muestras correlacionadas para Facebook.

Prueba de muestras relacionadas									
Diferencias relacionadas									
		Med	Desviac. Típica	Error típico de la media	95% Inter.confianza para la dif.		t	gl	Sig. (bilateral)
					Inf.	Sup.			
Par 1	Us - Sist2	,002	,07913	,02502	-,05461	,05861	,080	9	,938

Tabla 5. Anova y Correlaciones para Hotmail.

ANOVA						Correlaciones de muestras relacionadas				
Sistema2										
	Suma cuad.	gl	Mediacuad.	F	Sig.		N	Correl	Sig.	
Inter-grupos	,171	3	,057	4,090	,067	Par 1	Usuario y Sistema2	10	,713	,021
Intra-grupos	,084	6	,014							
Total	,255	9								

Tabla 6. Prueba de muestras correlacionadas para Hotmail.

Prueba de muestras relacionadas									
Diferencias relacionadas									
		Med	Desviac. Típica	Error típico de la media	95% Inter.confianza para la dif.		t	gl	Sig. (bilateral)
					Inf.	Sup.			
Par 1	Us - Sist2	,073	,11795	,03730	-,01138	,15738	1,957	9	,082

Tabla 7. Anova y Correlaciones para la Página de la Universidad.

ANOVA					
Sistema2					
	Suma cuad.	gl	Mediacuad.	F	Sig.
Inter-grupos	,098	5	,020	4,421	,087
Intra-grupos	,018	4	,004		
Total	,115	9			

Correlaciones de muestras relacionadas				
		N	Correl	Sig.
Par 1	Usuario y Sistema2	10	,477	,163

Tabla 8. Prueba de muestras correlacionadas para la página de la Universidad.

Prueba de muestras relacionadas									
Diferencias relacionadas									
		Med	Desviac. Típica	Error típico de la media	95% Inter.confiabledad para la dif.		t	gl	Sig. (bilateral)
					Inf.	Sup.			
Par 1	Us - Sist2	,002	,14274	,04514	-,10011	,10411	,044	9	,966

procesamiento, tal como lo encontraron en otras investigaciones de la misma índole [9].

La calidad en los productos de software es una realidad para el progreso tecnológico, la cual está en la búsqueda de un permanente perfeccionamiento de sus tareas [2], con el fin de generar soluciones a las grandes necesidades que presenta la ingeniería de software. Este proyecto define la forma en la cual los usuarios finales adoptan el concepto de calidad desde perspectivas disímiles y difícilmente parametrizables, dada la subjetividad del proceso, haciendo que la evaluación desde éste punto de vista, no sea completamente estándar[1], y de esta manera la investigación arroja unos resultados en términos de definiciones sobre las características que el usuario final busca como sinónimo de calidad, encontrándose que se centran en contenidos completos y actualizados, una interfaz gráfica agradable y bien distribuida, de fácil navegabilidad y enmarcado con parámetros de seguridad del sitio, resultados parecidos a los obtenidos por Chang [12] y donde la lógica difusa entra a generar un modelo matemático para la evaluación de cada una de esas variables y la generación de una calificación total.

En la revisión bibliográfica se observó que no existe teoría sobre la definición de atributos para valoración de la calidad externa de sistemas Web

dada la subjetividad del proceso y por ende se adoptó un desarrollo estadístico para la generación de dichos atributos, al igual que se hizo en la investigación adelantada por Yang [8], con el fin de corroborar los ítems de evaluación planteados en la presente investigación.

Por último, es importante resaltar el interés y la preocupación de la industria del software por perfeccionar éste tipo de temas, junto con la creación y generación de nuevos métodos y herramientas y en la formulación de nuevas métricas, las cuales son utilizadas en cada una de las fases de construcción del ciclo de vida del software, donde la lógica difusa y la inteligencia Artificial forman parte de un grupo selecto de herramientas para consolidar dicha área, como lo define Akingbehin [3] y otros autores.

Referencias

1. Calero, Coral; Moraga, María; Piattini, Mario. (2008). Handbook of research on web information system quality. Editorial Advisory Board – information Science Reference. New Zealand
2. Gall, C.; Lukins, S; Etzkorn, L; Gholston, S; Farrington, P; Utley, D; Fortune, J; Virani, S.(2008). Semantic software metrics computed from

- natural language design specifications. *Software, IET – IEEE*. Vol. 2 No. 1. pp. 17 – 26.
3. Akingbehin, Kiumi. (2009). Taguchi Smaller-the-Best Software Quality Metrics. 10th ACIS International Conference on Software Engineering, Artificial Intelligences, Networking and Parallel/Distributed Computing. pp. 585-588.
 4. Heck, Petra; Klabbers, Martijn; Eekelen Marko van. (2010). A software product certification model. *Software Quality Journal*, vol. 18 No. 1. - Springer Netherlands. pp. 37-55.
 5. Hofman, Radoslaw. (2009). Software Quality Perception. *Advanced Techniques in Computing Sciences and Software Engineering*. Springer. pp. 31 – 39.
 6. Pressman, Roger. (2008) *Ingeniería de Software, Un Enfoque Práctico*. 6 Edición. Editorial McGraw-Hill. Barcelona.
 7. López, Daniela; Agüero, Martín. (2007). Aplicación de Métricas Categóricas en sistemas de lógica difusa. *Revista IEEE América Latina*. Vol. 1. No. 5. pp. 55-61.
 8. Yang, Bo; Yao, Lan; Huang, Hong-Zhong. (2007). Early Software Quality Prediction Based on a Fuzzy Neural Network Model. *Third International Conference on Natural Computation (ICNC 2007) – IEEE*. pp. 760 – 764.
 9. Ruiz, Gustavo; Peña, Alejandro; Castro, Carlos; Alaguna, Ángela; Areiza, Luz y Rincón, Modelo de Evaluación de Calidad de Software Basado en Lógica Difusa, Aplicada a Métricas de Usabilidad de Acuerdo con la Norma ISO/IEC 9126. *Avances en Sistemas e Informática*. Vol. 3 No. 2 pp. 25–29.
 10. Morales Luna Guillermo. (2002). Introducción a la lógica difusa. *Centro de Investigación y Estudios Avanzados del IPN*. pp. 1 - 12
 11. Fernández, Carlos; Fernández, Ignacio; Moya, David. (2008). Valoración de inmuebles mediante técnicas de lógica difusa. *Universidad Complutense de Madrid*. Tesis de Maestría.
 12. Chang, Ching-Liang. (2009) *Fuzzy-logic-based programming*. Editorial World Scientific. *Advances in Fuzzy Logic. Applications and Theory*. Vol. 15.