



Hacia una certificación de la calidad sistémica en los sistemas de software en Venezuela

ANNA GRIMÁN¹

LUIS MENDOZA²

MARÍA PÉREZ³

MARYOLY ORTEGA⁴

^{1,2,3} Universidad Simón Bolívar, Dpto. de Procesos y Sistemas, LISI

¹ agriman@usb.ve

² lmendoza@usb.ve

³ movalles@usb.ve

⁴ Laboratorio de Computación, Universidad Ezequiel Zamora,

⁴ mortega@reacciun.ve

Resumen

Actualmente, Venezuela no cuenta con un modelo de especificación de la calidad que permita certificar a sus organizaciones desarrolladoras de sistemas de software (18). Ante esta situación, el Grupo de Investigación del Laboratorio de Investigación en Sistemas de Información (LISI) de la Universidad Simón Bolívar desarrolló el *MOdelo Sistémico de CALidad* (MOSCA), el cual trabaja con dos perspectivas: Producto y/o Proceso. MOSCA tiene una estructura jerárquica de cuatro niveles, cuenta con quinientas ochenta y siete (587) métricas para su operacionalización y con un algoritmo que parte de la medición de la calidad sistémica. Esta investigación tiene como propósito presentar las aplicaciones de MOSCA en un conjunto de empresas venezolanas desarrolladoras de sistemas de software, como primer paso en el avance hacia la certificación del software nacional. Se utilizó el método de Análisis de Características por Estudio de Caso para hacer las correspondientes evaluaciones. A través de los estudios, se comprobó la efectividad de MOSCA como Modelo de Certificación. En conclusión, MOSCA apoya la administración de la calidad del software al establecer un marco de referencia que ubica en un “nivel de calidad sistémica” a la organización y a sus productos y, con ello, se brinda orientación sobre actividades de planificación y control que conlleven a aumentar la calidad del software a los niveles deseados.

Palabras clave: Certificación, calidad sistémica, sistemas de *software* en Venezuela, calidad del proceso y calidad del producto.



Abstract

Currently, in Venezuela, a specification model for quality, which allows certifying the software systems development organizations, does not exist (18). In consequence, the Laboratorio de Investigación en Sistemas de Información (LISI) research group of Simón Bolívar University, has developed a Quality Systemic Model (known as MOSCA), which works on two bases: Product and/or Process perspectives. MOSCA has a four-level hierarchic structure, five hundred eighty and seven (587) metrics to operate, and one algorithm which begins with the systemic quality measurement. The purpose of this research is to present the MOSCA applications in a set of software systems development venezuelan enterprises as a first step to achieve the National Software Certification. The Feature Analysis Case Study was used to evaluate every one. In those studies, the MOSCA effectiveness as a certification model was proved. As a conclusion, MOSCA supports the Software Quality Management by establishing a framework that qualifies the organization and its products according to a "systemic quality level". In this way, it's possible to guide planning and control activities, in order to improve the software quality to achieve the expected levels.

Key words: Certification, systemic quality, software systems in Venezuela, process quality and product quality.

Introducción

La calidad del software es la concordancia de los requisitos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente (19).

A fin de evaluar y garantizar la calidad de los sistemas de software, no existen métricas estandarizadas y aplicables universalmente; es por ello, que las organizaciones deben seleccionar métricas y realizar mediciones basadas en el conocimiento y circunstancias locales (22). Este proceso es complejo y muchas veces arroja como resultado la necesidad de mejorar los niveles de calidad encontrados. Para estructurar las causas, consecuencias y posibles soluciones, se hace necesario recurrir al Enfoque de



Sistemas como una metodología de cambio (23) que permita comprender que todos los elementos (internos y externos) están interconectados y guardan una relación de afectación hacia la calidad.

En este sentido, el mejoramiento del sistema no sólo implicaría asegurar que éste opere de acuerdo a sus expectativas; sino además, trazar las desviaciones e investigar cómo se puede mejorar (23). Se puede decir, entonces, que ante la necesidad de un modelo de certificación de la calidad de los sistemas de software en Venezuela, es imprescindible que éste no sólo precise un nivel de calidad, sino que además indique los puntos a mejorar para aumentar los niveles actuales y permita hacer recomendaciones de cómo hacerlo; es decir, desde un enfoque sistémico.

Para esta investigación, primeramente se presentan los conceptos de la Calidad Sistémica y se describe el MOdelo Sistémico de CALidad (MOSCA) como antecedentes necesarios para su análisis. Seguidamente se muestran las diversas aplicaciones de este modelo a un conjunto de empresas desarrolladoras de software nacional desde sus dos perspectivas: Producto y/o Proceso indicando su contexto, la(s) perspectiva(s) y categoría(s) evaluada(s), el tipo de software y sus resultados. Por último, se presentan las conclusiones.

Calidad sistémica

La calidad del software es un concepto multidimensional que no se puede definir de forma simple. Clásicamente, la noción de calidad implica que el producto desarrollado cumple su especificación (22). La calidad total no se debe entender como la suma de calidades parciales, como es presentada por casi todos los autores (4, 19). No tiene ningún sentido diseñar un sistema altamente eficiente si no se utiliza; así como no tiene ningún sentido diseñar un sistema muy efectivo si no es factible por los recursos y/o a los apremios del tiempo (16). Es por ello que se debe considerar los recursos humanos, tecnológicos, financieros y de tiempo en cualquier diseño para el mínimo cumplimiento de un proceso eficaz, sin la coerción de la eficacia del producto. Si estas relaciones de la calidad no son consideradas en el diseño, la calidad global podría ser pobre (2).

Callaos y Callaos proponen un concepto de Calidad Sistémica del Software en el cual están involucrados tanto las características internas como el contexto organizacional, lo que genera un enfoque sistémico del concepto de calidad del software (2). Este enfoque es considerado también por Dromey (4), y particularmente reforzado por Voas (24), cuando se refiere al Triángulo de la Certificación de la Calidad del Software (Proceso, Personas y Tecnología).

La definición de Calidad Sistémica para los sistemas de software propuesta por Callaos y Callaos (2) se fundamenta en dos perspectivas: Producto y/o Proceso. Combinando el nivel de calidad de las características internas (aspecto interno) y el nivel de calidad del contexto organizacional (aspecto contextual), asociadas a la visión del cliente y/o usuario, por lo que se obtienen ocho (8) dimensiones (Figura N° 1).

Estas dimensiones se justifican porque en el contexto de un proyecto está presente tanto el diseño interno del producto como su satisfacción cuando está en uso. Igualmente, está presente el uso óptimo de recursos durante la construcción del sistema, así como el nivel de alineación del proceso con las metas organizacionales.

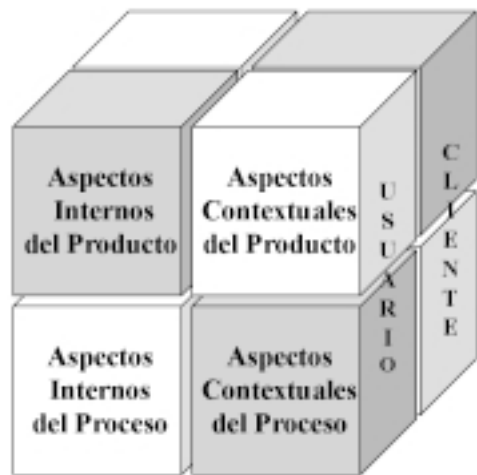


FIGURA N° 1
**MATRIZ GLOBAL DE LA CALIDAD
SISTÉMICA, ADAPTADO DE (2)**



Según Callaos y Callaos (2), la calidad global no es la suma de las calidades parciales, sino el compromiso entre todo el conjunto de calidades que conlleve a un óptimo global con cierto sacrificio de los óptimos parciales. Una calidad parcial se entiende como aquella que refleja los aspectos del Producto o del Proceso en la Matriz Global de la Calidad Sistémica.

La certificación, entonces, se apoyará en un modelo de especificación de la calidad que se apoye en la definición y principios de Calidad Sistémica antes dados.

Modelo Sistémico de Calidad - MOSCA

El proceso de construcción de un modelo es iterativo e incremental (8), razón por la cual el Laboratorio de Investigación de Sistemas de Información de la Universidad Simón Bolívar (LISI-USB) ha desarrollado dos versiones de un Modelo Sistémico de Calidad (MOSCA). La primera versión (12, 15 y 17) contempla las dos perspectivas: Producto y/o Proceso. La segunda versión (9) contempla tres perspectivas: Producto, Proceso y/o Humana, la cual es una investigación en progreso.

Tomando en cuenta la calidad del producto y la calidad del proceso, se desarrolló el **Modelo Sistémico de Calidad de software (MOSCA)**, por el LISI-USB (12), que integra el modelo de calidad del producto (15) y el modelo de calidad del proceso de desarrollo (17), y está soportado por los conceptos de Calidad Sistémica.

En la Figura Nº 2 se muestra su estructura en su primera versión. En ésta se observan sus dos perspectivas: *Producto y/o Proceso*. De acuerdo con los objetivos previstos en la evaluación del modelo MOSCA se puede tomar la perspectiva del *Producto*, la del *Proceso*, o ambas inclusive. La primera de ellas se utiliza para evaluar sistemas de software ya elaborados, mientras que el segundo se emplea cuando además, se requiere evaluar el proceso de su desarrollo.

En cuanto al producto, este modelo plantea, con base en las seis (6) características de calidad del estándar internacional ISO/IEC 9126 (6), un conjunto de categorías, características y métricas asociadas que miden la

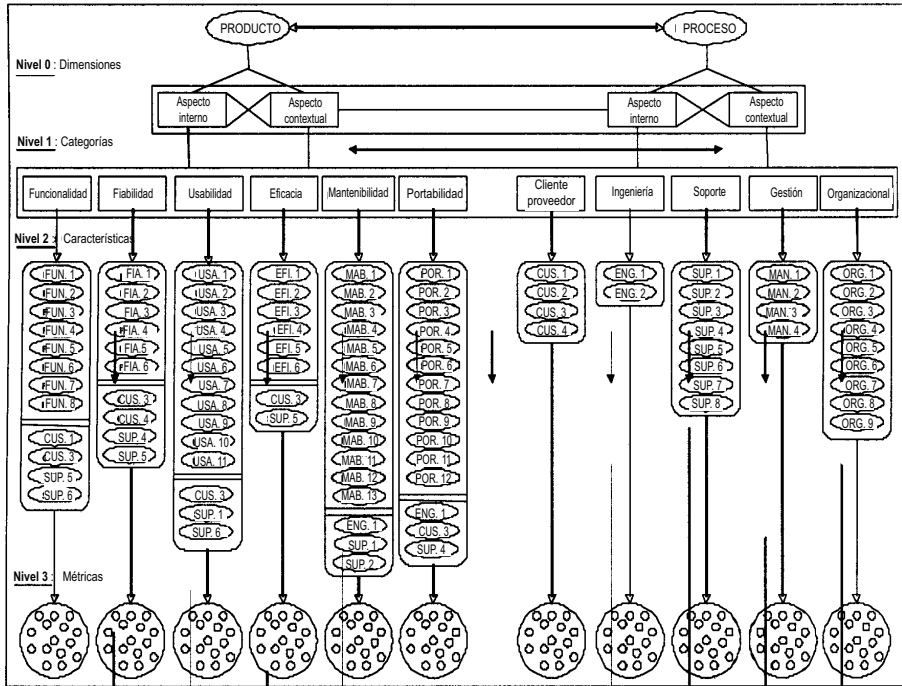


FIGURA Nº 2

DIAGRAMA DEL MODELO SISTÉMICO DE CALIDAD - MOSCA (13).

calidad de un producto de software con un enfoque sistémico y hacen del modelo un instrumento de medición de gran valor, ya que cubre todos los aspectos imprescindibles para medir directamente la calidad del producto de software.

En cuanto al proceso, este modelo plantea en base a los cinco (5) subprocesos del estándar internacional ISO/IEC 15504 (7), un conjunto de categorías, características y métricas asociadas que miden la calidad de un proceso de software con un enfoque sistémico.

A continuación se explican los niveles que conforman el modelo MOSCA en su primera versión:



Nivel 0: Dimensiones. Las cuatro dimensiones propuestas por MOSCA son: *Aspectos Internos y Contextuales del Proceso*, *Aspectos Internos y Contextuales del Producto*.

Nivel 1: Categorías. Este nivel contempla once (6) categorías, de las cuales seis (6) pertenecen al *Producto* y cinco (5) al *Proceso* de desarrollo. En las tablas 1 y 2 se muestran las definiciones de las categorías pertenecientes al *Producto* y al *Proceso*, respectivamente.

TABLA N° 1
CATEGORÍAS DE LA PERSPECTIVA DEL PRODUCTO DE MOSCA (13)

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

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 del producto y del proceso.



Nivel 3: Métricas. Cada característica posee una serie de métricas que están relacionadas con las cualidades que se desean evaluar del producto de software. Dichas métricas permiten evaluar la calidad del producto.

TABLA N° 2

MOSCA - CATEGORÍAS PARA LA PERSPECTIVA DEL PROCESO (13)

Categoría	Definición
Cliente - Proveedor (CUS)	Está conformada por los procesos que impactan directamente al cliente, apoya el desarrollo y la transición del software hasta el cliente, y provee la correcta operación y uso del producto o servicio de software.
Ingeniería (ENG)	Consiste en procesos que directamente especifican, implementan o mantienen el producto de software, su relación con el sistema y su documentación.
Soporte (SUP)	Consta de procesos que pueden ser empleados por cualquiera de los procesos (incluyendo los de soporte) en varios niveles del ciclo de vida de adquisición.
Gestión (MAN)	Consiste en procesos que contienen prácticas de naturaleza genérica, que pueden ser utilizadas por cualquier personaje que dirija algún tipo de proyecto o proceso, dentro de un ciclo de vida primario.
Organizacional (ORG)	Está relacionado con procesos que establecen las metas comerciales de la organización y desarrollan bienes (valores) de proceso, producto y recurso, que ayudarán a las organizaciones a alcanzar sus metas en los proyectos.

MOSCA consta de un total de quinientas ochenta y siete (587) métricas. Además, cuenta con un algoritmo para medir la calidad sistémica (13). El algoritmo contempla tres (3) fases:

Fase 1: Calidad del producto de software con un enfoque sistémico.



Fase 2: Calidad del proceso de desarrollo de software con un enfoque sistémico.

Fase 3: Integración de las mediciones de los submodelos de la calidad del producto y la calidad del proceso.

Cabe acotar que, según el método propuesto para aplicar el modelo de calidad en su Perspectiva Producto (12), no todas las características y métricas tienen que ser usadas, sino que el modelo debe ser adaptado dependiendo de las características que se desean del producto, en virtud del dominio de aplicación del software (13).

Para la estimación de la Calidad del Producto y del Proceso, el Algoritmo de MOSCA (13) indica que una categoría es satisfecha si cumple con el mínimo requerido de sus características satisfechas, el cual varía dependiendo de la categoría. Si el setenta y cinco por ciento (75%), como mínimo, de las métricas asociadas se encuentran dentro de los valores óptimos (mayor o igual a 4) para cada una de sus características inmersas en las categorías evaluadas, entonces se puede señalar que tal característica ha sido satisfecha.

Se debe tomar en cuenta que utilizando un proceso formal de evaluación de sistemas de software se reduce la posibilidad de adquirir productos inadecuados o innecesarios y, por tanto, se sugiere usar métodos de evaluación que permitan medir factores claves en el producto a adquirir (21). MOSCA ha sido aplicado en varias organizaciones desarrolladoras de sistemas de software en Venezuela a lo largo de tres años, estimándose con él la Calidad Sistémica del producto, del proceso o ambos. A fin de tener una visión global del contexto de aplicación de este modelo, se presenta a continuación un resumen de cada estudio de acuerdo al ámbito del diagnóstico (Producto, Proceso o ambos). Es importante señalar que durante este período, MOSCA ha sufrido mejoras como resultado del ciclo de aprendizaje.

1. MOSCA Producto_Software Bancario (14)

La empresa donde se realizó la evaluación pertenece al sector bancario venezolano, ubicándose dentro del área de servicios. Fueron seleccionados



dos proyectos de la Gerencia de Riesgos de la Vicepresidencia de Sistemas Bancarios, ubicada en la Vicepresidencia Ejecutiva de Tecnología. Estos Sistemas fueron “Monitoreo especial” y “Valoración”.

La importancia de esta investigación radica en el interés creciente que existe dentro de las organizaciones desarrolladoras de sistemas de software en obtener certificaciones de sus productos y la falta de mecanismos que existen para obtenerlos, ya que para vender el software (como producto) necesitan plegarse a lineamientos de calidad que permitan mostrarle a las empresas compradoras que su producto es bueno. Es por ello que sólo se trabajó con la *Perspectiva del Producto*.

Se utilizó la *Perspectiva Producto de MOSCA v.1*. Los datos fueron analizados tomando en cuenta (14): a) las características y subcaracterísticas de los dos productos de software seleccionados por el interesado en la evaluación; b) la comparación entre los dos productos de software y, por último; c) un análisis del comportamiento de las dimensiones de la calidad para ambos productos. Con respecto al producto “Monitoreo Especial” (Producto P1), se evaluaron las Categorías **Funcionalidad y Mantenibilidad** con todas sus características y la Categoría de **Eficiencia**, a excepción de la Característica de “Procesos relacionados con la eficiencia”. En cuanto al producto “Valoración” (Producto P2), sólo se evaluó la Categoría **Funcionalidad** con todas sus características, por ser crítica para el ente evaluado.

La Figura 3 muestra los resultados en cuanto a los requerimientos de Calidad establecidos por el cliente para el producto “Monitoreo Especial” (Producto P1): **En la Categoría Funcionalidad**: en cuanto a los Aspectos Internos del Producto, el setenta y cinco por ciento (75%) de las características presentaron un nivel óptimo, por lo que este aspecto debe ser cuidado, ya que al encontrarse dentro del nivel mínimo requerido por MOSCA puede repercutir sobre futuros cambios en la Funcionalidad del producto (13). En cuanto a los Aspectos Contextuales del producto, ésta presentó un nivel muy óptimo (noventa y cuatro coma cuatro por ciento - 94,4%).

En la Categoría Eficiencia: se observó el cumplimiento en un cien por ciento (100%) de todas las características relacionadas con los Aspectos Internos y Contextuales del Producto.

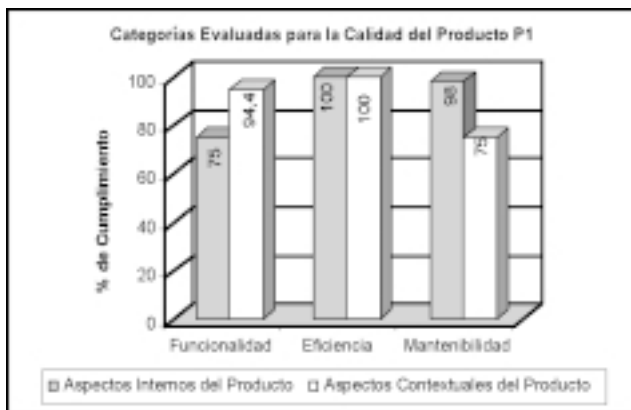


FIGURA Nº 3
RESULTADOS DE LA EVALUACIÓN DE LAS CATEGORÍAS
PARA LA CALIDAD DEL PRODUCTO P1 (14)

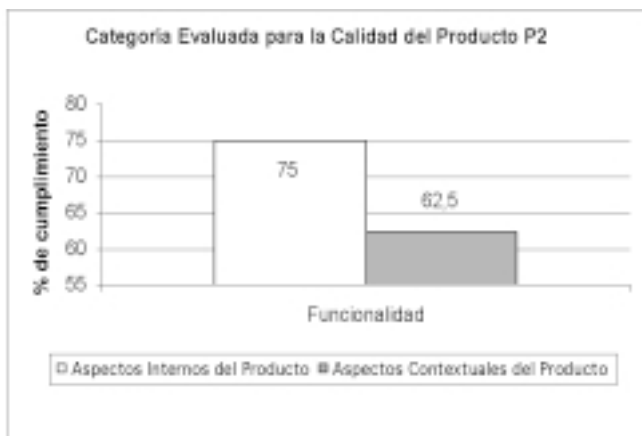


FIGURA Nº 4
RESULTADOS DE LA EVALUACIÓN DE LA CALIDAD
EN LA CATEGORÍA FUNCIONALIDAD DEL PRODUCTO P2 (14)

En la Categoría Mantenibilidad: el Aspecto Interno del producto presentó un nivel muy óptimo (noventa y ocho por ciento - 98%) debido a la repercusión de las características de Capacidad de Análisis y de Estabilidad



del Producto de software; en cuanto a los Aspectos Contextuales del Producto, ésta presentó un nivel óptimo de setenta y cinco por ciento (75%).

El producto “Valoración” (Producto P2) no presentó un nivel óptimo (Figura N° 4). **En cuanto a la Categoría Funcionalidad**, se puede observar que presentó un setenta y cinco por ciento (75%) de cumplimiento para los aspectos internos del Producto debido a que la característica de Interoperabilidad (14) presenta mayor debilidad; esto permitió entender que las mejoras en ésta repercuten en la Funcionalidad. El usuario pudo detectar que existían funcionalidades en otros productos de software que favorecían la Funcionalidad del Producto P2. Se obtuvieron niveles óptimos relacionados con los Aspectos Contextuales del Producto (setenta y cinco por ciento - 75%).

Si se toman en cuenta todos los valores de las métricas (no se establecen valores mínimos, donde un cien por ciento –100%– equivale a cinco –5– en todas las métricas), el Producto “Monitoreo Especial” (Producto P1) presentó una puntuación mayor al producto “Valoración” (Producto P2), lo que indica que éste último sólo tomó valores óptimos y no sobresalientes (14).

Después de haber observado el comportamiento del modelo en los dos productos diferentes, se concluyó que el modelo permite detectar fortalezas y debilidades en los productos (14) y facilita de una manera formal, información válida que puede ser utilizada para las mejoras de los productos. Con el uso del Modelo se detectaron los Aspectos Internos del Producto que influyen en sus Aspectos Contextuales. Gracias al establecimiento en el modelo de los requerimientos sobre las métricas (mayores o igual a tres –3), se pueden identificar niveles inferiores de rendimiento de los productos que de otra forma sería difícil de detectar (13).

2. MOSCA_Proceso (1)

La unidad de análisis para esta investigación es el sector industrial venezolano relacionado con el desarrollo de sistemas. En consecuencia, la muestra a medir comprende dos tipos de organizaciones: la **Organización A**, la cual se caracteriza por ser una empresa pequeña, especializada en el desarrollo de soluciones de automatización para la industria de telecomunicaciones. Su campo de acción se concentra en el desarrollo de productos



y servicios para la pequeña y mediana empresa, a través de paquetes de software abiertos. La **Organización B**, es una empresa de gran magnitud especializada en el área financiera, que se ha consolidado con el paso de los años. Su campo de acción se limita a una unidad departamental.

Se utilizó la *Perspectiva Proceso de MOSCA v.1* para la evaluación de todas sus categorías: **Cliente-Proveedor (CUS)**, **Ingeniería (ENG)**, **Soporte (SUP)**, **Gestión (MAN)** y **Organizacional (ORG)**.

Como resultado de su aplicación (Figura 5), se puede concluir que la **Organización B** ha implementado las categorías **Cliente-Proveedor** e **Ingeniería**, ya que el nivel de ambas está por encima de setenta y cinco por ciento (75%). Por su parte, las categorías **Soporte** y **Gestión** (13), a pesar de estar implementadas presentan ciertos problemas contextuales, y la categoría **Organizacional** (13) no está implementada. La **Organización A** ha implementado las categorías **Cliente-Proveedor**, **Ingeniería** y **Gestión** pero no de manera efectiva; las categorías de **Soporte** y **Organizacional** no están implementadas. La **Organización A** carece de Calidad en el proceso y la **Organización B** presenta una calidad del Tipo Básica.

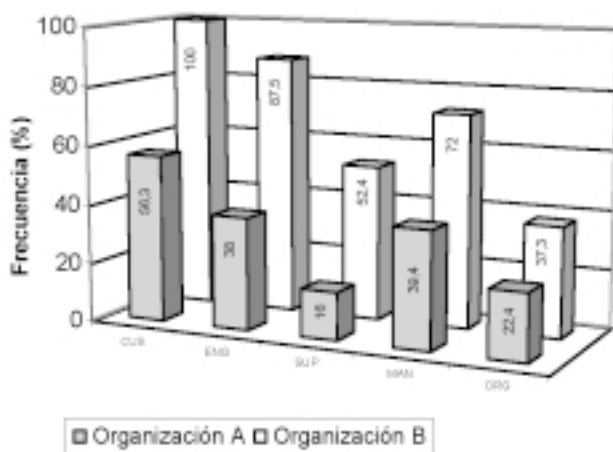


FIGURA N° 5
**RESULTADOS EN LAS CATEGORÍAS DE LA PERSPECTIVA PROCESO
 EN LA ORGANIZACIÓN A Y B (1)**

La calidad del proceso Básica permite asegurar que existen procesos estándares para el desarrollo que permiten asegurar que los Aspectos Internos y contextuales asociados al ciclo de vida primario se están cumpliendo (1). La calidad de la *Perspectiva Proceso*, variable observada y analizada, presentó el comportamiento que se esperaba tanto por parte de los participantes de las muestras como del grupo de investigación. La ausencia de la calidad del proceso en la **Organización A** se debe a que todo el proceso de desarrollo, como ellos mismos explican, se basa en el factor de la experiencia de quienes participan en los procesos o actividades a realizar. Por lo que se concluye que la aplicación de MOSCA fue efectiva (1).

3. MOSCA Producto_Proceso (10)

Se evaluó el modelo en dos empresas desarrolladoras de software distintas: la **Empresa A** es una compañía cuyo objetivo es brindar soluciones basadas en web a distintas empresas, tanto nacionales como internacionales. La **Empresa B** es un grupo conformado por siete empresas dedicadas a las diferentes áreas de tecnología de información. Entre los servicios que presta se encuentran “outsourcing” operativo en el área de informática o segmentos de los mismos, servicios de implantación tecnológica, infraestructura computacional, entre otras.

En ambas empresas se seleccionó un producto de software emblemático y que, además, fuese desarrollado recientemente. En el caso de la **Empresa A** se seleccionó un *site web* de una empresa venezolana y, con respecto a la **Empresa B**, se escogió una herramienta que permite la administración y control de la ejecución de los procesos funcionales, a través de la configuración de los aplicativos instalados en el producto.

Se evaluaron las dos perspectivas: *Producto y Proceso de MOSCA v.1*. Dentro de la perspectiva *Producto* se evaluó el comportamiento de las categorías: **Funcionalidad**, **Usabilidad** y **Mantenibilidad**. En la vista de *Procesos*, se evaluaron **todas sus categorías**.

Los resultados para la **Empresa A** se muestran en la Figura 6: en cuanto a la Calidad de la *Perspectiva Producto* se tiene que, siguiendo con el algoritmo de MOSCA, presenta un nivel de Calidad Intermedio ya que cumple con dos



FIGURA Nº 6
RESULTADOS EN LAS CATEGORÍAS DE LAS PERSPECTIVAS PRODUCTO Y PROCESO EN EMPRESA A (10)

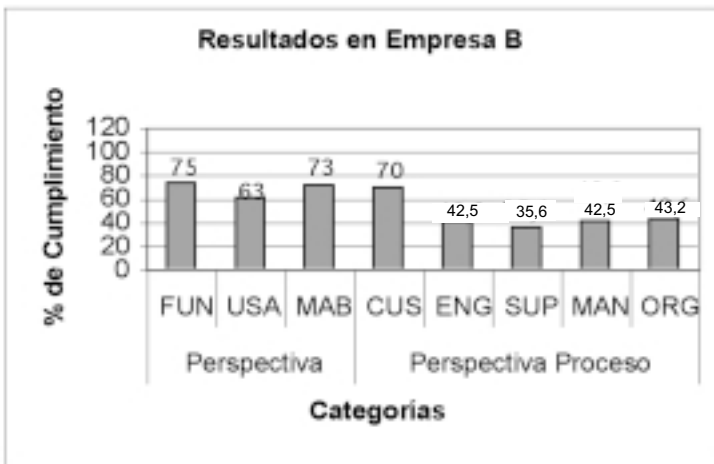


FIGURA Nº 7
RESULTADOS EN LAS CATEGORÍAS DE LAS PERSPECTIVAS PRODUCTO Y PROCESO EN EMPRESA B (10)



(2) de las tres (3) categorías instanciadas (**Funcionalidad y Usabilidad**). No cumple con la **Mantenibilidad**. La *Perspectiva del Proceso* tiene un nivel de Calidad Básica ya que cumple únicamente con las categorías **Ciente-Proveedor** e **Ingeniería**. El nivel de *Calidad Sistémica* es Básico.

En cuanto a la **Empresa B** (Figura N° 7): la calidad a nivel de la *Perspectiva Producto* es Nula, ya que no fueron altamente satisfechas las categorías (10). A nivel de la *Perspectiva Proceso* es Nula ya que no cumple con la categoría **Ciente-Proveedor** ni con la categoría **Ingeniería**, que son las mínimas para optar por un nivel de calidad básico. Por tanto, la *Calidad Sistémica* es Nula ya que tanto el producto como el proceso de desarrollo del mismo presentan grandes deficiencias (10).

A través de este estudio se pudo constatar la influencia de algunas características de la *Perspectiva Proceso* sobre las Categorías de la *Perspectiva Producto*. Después de aplicar MOSCA v.1 se concluyó que constituye una herramienta efectiva de análisis y estimación, debido a que se analizan aspectos del Producto, del Proceso y su relación en el medio ambiente. Las empresas evaluadas mostraron una alta aceptación cuando se aplicó el modelo a cada una de ellas y sobre todo cuando se les presentaron los resultados obtenidos, sus respectivas explicaciones y los puntos a considerar para mejorar la calidad sistémica (10).

4. MOSCA v.1.1. *Producto_Software Educativo* (3)

En este estudio de caso se seleccionó un conjunto de cuatro (4) títulos, dos en español y dos en inglés, para distintas áreas de la educación utilizados a nivel de edad escolar y adolescente, en la Unidad Educativa Institutos Educativos Asociados (IEA), Caracas, Venezuela.

Cuando se hace referencia a calidad de software educativo se requiere un producto que satisfaga tanto las expectativas de los docentes como las de los alumnos, a un menor costo, libre de errores y cumpliendo con ciertas especificaciones instruccionales y tecnológicas (3). Por ello, se partió de MOSCA v.1 en su *Perspectiva Producto* y se amplió de acuerdo con los requerimientos particulares de calidad del software educativo, tomando en cuenta no sólo los aspectos técnicos del producto, sino el



diseño pedagógico y los materiales de soporte didáctico; esta ampliación generó MOSCA v.1.1 (3).

En la Figura N° 8 se muestran los porcentajes alcanzados por los cuatro (4) software educativos, considerando las categorías: **Funcionalidad, Usabilidad y Fiabilidad** (3).

Resumiendo, se tiene que dos (2) de los cuatro (4) software evaluados presentan un nivel de calidad Avanzado; ellos son: “Los aventureros 3er. Grado” y “Trampolín tercer ciclo” al presentar niveles de cumplimiento por encima del setenta y cinco por ciento (75%) en las tres categorías. El software “Smart Start in English (Hable Inglés ¡ya!)” presenta nivel de Calidad Básica, debido a que la categoría Usabilidad no fue satisfecha. El software “Kids English” presenta nivel de calidad Nula, debido a que la categoría Funcionalidad y Usabilidad no fueron satisfechas.

MOSCA v.1.1 proporciona una eficiente herramienta de evaluación para el docente que tiene la necesidad de conocer el valor educativo de un software, su calidad y su uso en el ambiente educativo, como criterios imprescindibles para su adquisición (3). Es además, una excelente herramienta de evaluación para hacer ajustes durante el proceso de diseño o desarrollo

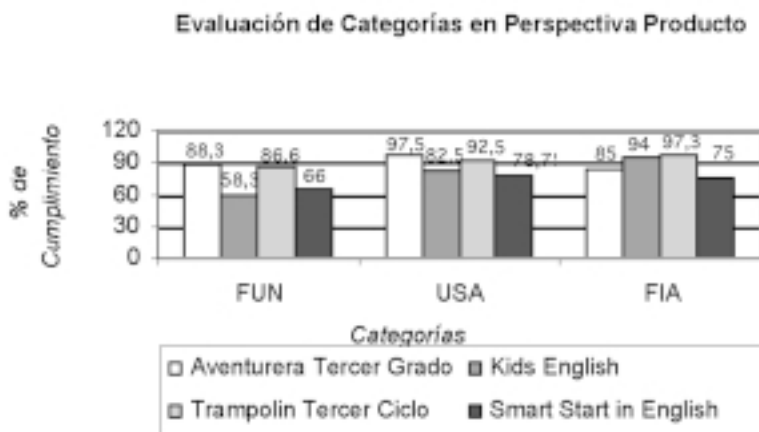


FIGURA N° 8
RESULTADOS EN LAS CATEGORÍAS DE LA PERSPECTIVA PRODUCTO (3).



de software educativo, de una evaluación de prueba, antes de la edición definitiva.

5. *MOSCA v.1.2. Producto_Simulación de Eventos Discretos (20)*

La investigación se desarrolló en una organización que presta servicios de consultoría en el área de logística de la industria petrolera. La complejidad conceptual de los sistemas logísticos de suministro, manejo, transporte y distribución de hidrocarburos en la industria petrolera, están caracterizados por procesos en los cuales interactúan una serie de factores que producen fenómenos probabilísticos y ocurrencias de eventos discretos (5).

En la búsqueda de trabajar con un sistema organizado que permita ordenar la información sobre Software de Simulación de Eventos Discretos (SSED) y que conduzca a obtener una herramienta que sea útil para evaluar este tipo de software en la industria petrolera, se evaluaron cuatro (4) paquetes sobre la base de la *Perspectiva del Producto* sin considerar la *Perspectiva del Proceso*, puesto que el interés de la organización es la evaluación y la selección de software preexistentes como productos terminados.

La gran cantidad de rasgos a evaluar en los SSED, sumada a la complejidad de los sistemas de logística para este tipo de industria, dirigió esta investigación a utilizar la *Perspectiva Producto de MOSCA v.1* con el objetivo de agrupar dichos rasgos de una forma coherente y fácil de manipular y entender. Los ciento treinta y un (131) rasgos identificados (20) se distribuyeron de la siguiente manera: ochenta y seis (86) para los criterios de la categoría **Funcionalidad**, treinta y nueve (39) para los criterios que conforman la categoría **Usabilidad** y seis (6) para la categoría **Eficiencia**, generando de esta forma a MOSCA v.1.2.

Los resultados guardan correspondencia con los objetivos propuestos y con la orientación metodológica diseñada en función de la naturaleza y fines del mismo. Por esta razón, se registraron logros de tipo cualitativo y procedimental, así como elementos cuantitativos para sustentar la selección de herramientas para la simulación de operaciones relacionadas con la logística de suministro, manejo, transporte y distribución de hidrocarburos (20).



La Figura N° 9 presenta los resultados obtenidos de la aplicación del modelo propuesto (MOSCA v.1.2) para la evaluación de los cuatro (4) SSED seleccionados.

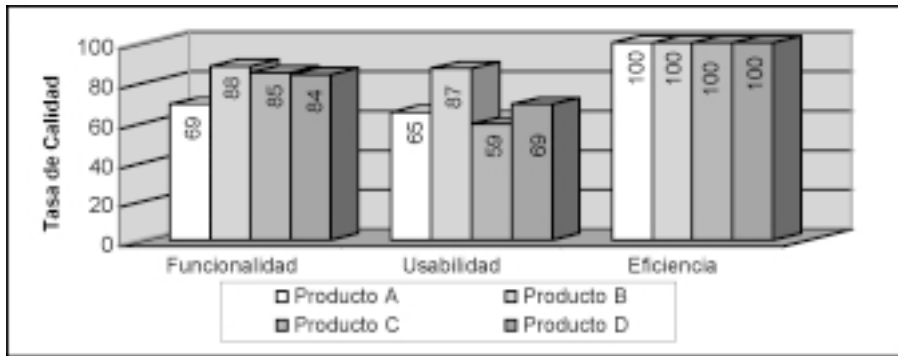


FIGURA N° 9
RESULTADOS DE LAS CATEGORÍAS EVALUADAS PARA LA CALIDAD DE LOS PRODUCTOS A, B, C Y D (20)

Los resultados reflejan que el modelo aplicado permitió diferenciar claramente las fortalezas y debilidades de estos SSED, apoyando efectivamente la toma de decisiones con base en las necesidades particulares de esta organización. Esta investigación permitió validar la aplicabilidad de MOSCA para la evaluación de los paquetes de simulación (20) porque se pudo adaptar a las necesidades particulares de este tipo de software, confirmando su carácter sistémico.

Conclusiones

Después de presentar las diversas aplicaciones del MOdelo Sistémico de CALidad (MOSCA), se probó su efectividad (11) para medir la Calidad de Productos de Software desde su perspectiva Producto y/o Proceso en las organizaciones venezolanas desarrolladoras de sistemas de software donde ha sido probado (1, 3, 10, 14, y 20). Además, se pudo constatar las relaciones entre la calidad del producto y del proceso a través de las categorías evaluadas.



Así mismo, MOSCA es una herramienta que soporta la Administración de la Calidad del Software en sus tres actividades: Aseguramiento de la Calidad, Planeación de la Calidad y Control de la Calidad, al establecer un marco de referencia que permite ubicar en un “nivel establecido” la calidad sistémica de sus productos. A partir de esto, se pueden propiciar actividades de planificación y control (según el diagnóstico de la evaluación) que conlleven a aumentar los niveles de calidad de los productos de software. Esto se realiza ya que en base a los resultados mostrados por las métricas asociadas a cada característica de calidad evaluada se les brinda una serie de recomendaciones u orientaciones a los usuarios-clientes de la organización que les permitirá mantener, controlar y/o mejorar los niveles de calidad.



Bibliografía

- (1) ÁLVAREZ, A. (2000). *Modelo para la Evaluación de la Calidad del Proceso de Desarrollo de Sistemas*. LISI - Universidad Simón Bolívar. Caracas-Venezuela.
- (2) CALLAOS, N. y CALLAOS, B. (1996). *Designing with Systemic Total Quality*. In International Conference on Information Systems. *International Institute of Informatics and Systemics*, Orlando, USA, pp. 548-560.
- (3) DÍAZ, G.; PÉREZ, M.; MENDOZA, L. y GRIMÁN, A. (2002). *Ampliación de MOSCA para la Evaluación de Software Educativo*. Modalidad: ORAL. LII Convención Anual de AsoVAC. Barquisimeto - Venezuela. Vol. 53, Sup. 1, pp. 361-362.
- (4) DROMEY, R. (1996). *Concerning the Chimera*. *IEEE Software*. Vol. 13, Nº 1, pp. 33-43.
- (5) FALCONER, D. y GUY, B. (1998). *Trully Optimal Offsites*. *The Chemical Engineer*, marzo, pp. 28-33.
- (6) ISO/IEC 9126-1.2 (1998). *Information Technology - Software Product Quality - Part 1: Quality Model*, ISO/IEC JTC1/SC7/WG6.
- (7) ISO/IEC TR 15504-2 (1998). *(E) Information technology - Software process assessment -Part 2: A reference model for processes and process capability*, ISO/IEC JTC 1/SC 7. Canadá.
- (8) LARMAN, C. (2003). *UML y Patrones*. Segunda edición, España: Editorial Pearson-Prentice Hall, p. 14.
- (9) MARTÍN, C. (2003). *Propuesta de Modelo Sistémico de Calidad de Software (MOSCA) en la Dimensión Usuario-Cliente*. LISI - USB y Universidad Católica Andrés Bello. Caracas -Venezuela.
- (10) MARTÍNEZ, J. (2001). *Modelo Sistémico de Calidad – MOSCA*. LISI - Universidad Simón Bolívar. Caracas-Venezuela.
- (11) MAYORCA, M. (2004). *Método de Evaluación Sistémica*. LISI - Universidad Simón Bolívar. Caracas-Venezuela.
- (12) MENDOZA, L.; PÉREZ, M. y ROJAS, T. (2001). "Modelo sistémico para estimar la calidad de los sistemas de software (MOSCA)". *ASOVAC, Acta Científica Venezolana*, Edo. Lara, Venezuela. Vol. 53, Nº 3, p. 435.



- (13) MENDOZA, L.; PÉREZ, M.; GRIMÁN, A. y ROJAS, T. (2002). *Algoritmo para la Evaluación de la Calidad Sistémica del Software*. Anales de las 2das. Jornadas Iberoamericanas de Ingeniería del Software e Ingeniería del Conocimiento (JIISIC 2002). Salvador - Brasil, pp. 1-11.
- (14) ORTEGA, M. (2000). *Modelo de Calidad del Producto del Software con un enfoque sistémico*. LISI - Universidad Simón Bolívar. Caracas-Venezuela.
- (15) ORTEGA, M.; PÉREZ, M. y ROJAS T. (2003). "Construction of a Systemic Quality Model for Evaluating a Software Product". *Software Quality Journal*. Indexada en el SCI. Vol. 11, pp. 219-242.
- (16) PÉREZ, M.; ROJAS, T.; ORTEGA, M. y ÁLVAREZ, C. (1999). *Toward Systemic Quality: Case study*. Porlamar, Venezuela: 4Th Squad Meeting.
- (17) PÉREZ, M.; MENDOZA, L.; ROJAS, T. Y GRIMÁN, A. (2001). *Systemic Quality Model for System Development Process: Case Study*, en Seventh Americas Conference on Information Systems AMCIS 2001. Boston - USA, agosto.
- (18) PÉREZ, M.; ROJAS, T. y ORTEGA, M. (2002). *Certificación de la Calidad Total Sistémica en el Proceso de Desarrollo de los Sistemas de Información*. Informe Técnico de Proyecto de Fonacit.
- (19) PRESSMAN, R. (2002). *Ingeniería del Software: un enfoque práctico*. Quinta edición, España: Editorial Mc Graw Hill, pp. 135.
- (20) RINCÓN, G.; PÉREZ, M.; ÁLVAREZ, M. y HERNÁNDEZ, S. (2003). *Modelo de calidad (MOSCA+) para evaluar software de simulación de eventos discretos*. 6º Workshop Iberoamérica de Ingeniería de Requisito y Ambientes Software IDEAS 2003. Asunción - Paraguay. Vol. 1, pp. 167-177.
- (21) ROJAS, T.; PÉREZ, M.; GRIMÁN, A.; ORTEGA, M. y DÍAZ, A. (2000). "Modelo de decisión para soportar la selección de herramientas CASE". *Revista de la Facultad de Ingeniería* de la UCV, Vol. 15, Nº 2, pp. 117-144.
- (22) SOMMERVILLE, I. (2002). *Ingeniería del Software*. Sexta edición, México: Editorial Addison Wesley, pp. 536 y 555.
- (23) VAN GIGCH, J. (1989). *Teoría General de Sistemas*. Segunda edición. México: Editorial Trillas, pp. 16 y 18.
- (24) VOAS, J. (1999). *Software Quality's Eight Greatest Myths, IEEE Software*. Vol. 16, Nº 5, September/October, pp. 740-745.