



**EVALUACIÓN DE LA CALIDAD DEL SOFTWARE  
EDUCATIVO BAJO EL ESTÁNDAR ISO 9126**

Mgr. JOSÉ ALBERTO CRISTANCHO

Instituto Universitario de Tecnología Región los Andes  
IUT-Táchira

**RESUMEN**

La calidad del software educativo puede ser valorada en función de su apego al cumplimiento de ciertas especificaciones técnicas; adicionalmente, es conveniente considera el diseño pedagógico que aborda su adecuación a las exigencias educativas. La Organización Internacional para la Estandarización creó el estándar ISO 9126 que establece, en seis categorías, las "pautas de normalización" referidas a los criterios de calidad; esto ha favorecido la consideración de otros aspectos que permiten superar los límites de un proceso de evaluación centrado exclusivamente en su "funcionalidad". Se propone aquí, evaluar la efectividad del software, utilizando tres de los seis parámetros referidos: funcionalidad, usabilidad y fiabilidad. La Universidad Simón Bolívar, a través de su Laboratorio de Investigación de Sistemas de Información (LISI-USB), ofrece un ajuste de las métricas descritas en el modelo ISO 9126, adecuadas a las necesidades propias del software que se pretende evaluar. Este aporte facilita la valoración del software y la "evaluación" del proceso de su inserción en escuelas e institutos educativos.

Palabras Claves: calidad, software educativo, efectividad, evaluación, funcionalidad, usabilidad, fiabilidad, Tic's.

**QUALITY EVALUATION OF THE EDUCATIONAL SOFTWARE  
UNDER ISO 9126 STANDARD**

**ABSTRACT**

The quality of educational software is valued in terms of its truthfulness to certain technical specifications, additionally, the pedagogical design is considered in relation to its adaptability to educational demands. The International Organization for Standardization created the ISO 9126 Standard that establishes, in six categories, "the outlines for normalization" dealing with quality criteria, this has favored the consideration of other aspects that allow to overcome the boundaries of an evaluation process focused exclusively on its "functionability." It is intended here, to evaluate the effectiveness of the software, using three of the six aforementioned parameters: functionability, useability, and fiability. The Simon Bolivar University through its Information System Research Laboratory (LISI-USB) offers a readjustment of the metrics described in the ISO 9126 Model adapted to the specific software needs that it is intended to evaluate. This contribution facilitates the software assessment and the process "evaluation" of the process of insertion in schools or educational institutions.

Key Words: Quality, Educational Software, effectiveness, evaluation, functionability, useability, fiability, TIC's.

## **EVALUACIÓN DE LA CALIDAD DEL SOFTWARE EDUCATIVO BAJO EL ESTÁNDAR ISO 9126**

Las Tecnologías de la Información y la Comunicación (TIC's) y los "softwares" de contenido, entre otros, hoy en día son elementos presentes en la educación; por tanto resulta indispensable garantizar que estas herramientas de apoyo sean de calidad, en particular, el software que soporte el proceso de enseñanza-aprendizaje. La calidad es un elemento multidimensional y contextual, por tanto, altamente cualitativo; sin embargo la tendencia mundial en cuanto a evaluación de la "calidad", se orienta a la "cuantificación" de los aspectos sometidos a consideración, con el fin de contribuir en la toma de decisiones dentro de un proceso de selección y adquisición de alternativas.

Respecto al software educativo, se requiere un producto que satisfaga tanto las expectativas de los docentes como de los usuarios, a un menor costo, libre de defectos, y que cumpla con ciertas especificaciones (Pressman, 2002). Por su parte Gros (2000), considera que la calidad del software educativo está determinada no sólo por los aspectos técnicos del producto, sino por el diseño pedagógico y los materiales de soporte. Este último aspecto constituye uno de los puntos álgidos y problemáticos a la hora de evaluar software educativo, ya que existen poco programas que ofrezcan un soporte didáctico.

Tradicionalmente la evaluación de software educativo se ha centrado en dos momentos:

1. Durante su utilización real por parte de los usuarios, para juzgar su eficiencia y los resultados que con él se obtienen, y
2. Durante el proceso de diseño y desarrollo, con el fin de corregir y perfeccionar el programa.

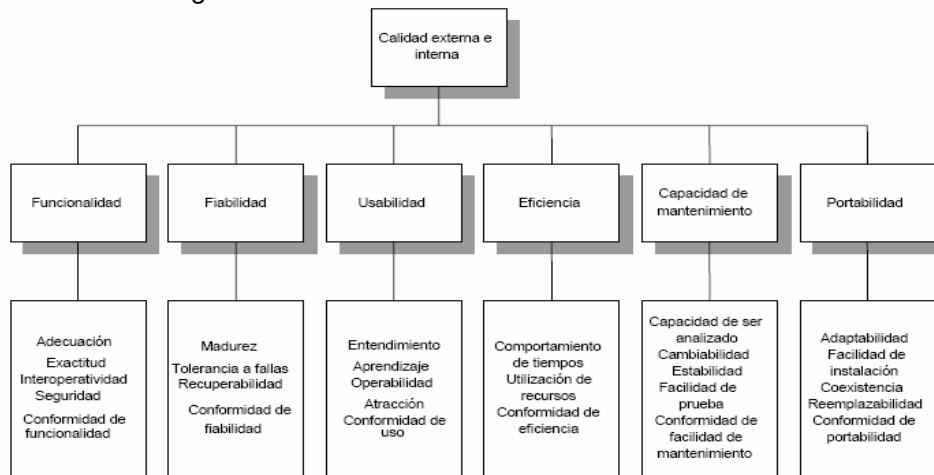
Para Callaos y Callaos (1993), la calidad del software no es algo que depende de una sola característica en particular, sino que obedece al compromiso de todas sus partes, valorándose la calidad del producto y la calidad del proceso. Es relevante valorar, que la funcionalidad de un producto ha sido la única manera de medir la calidad del software. Es así, que era necesaria la llegada de un modelo estándar que estableciera las características fundamentales de calidad del software y que ganara el consenso para su aplicación.

Por esta razón, el comité técnico de la ISO/IEC<sup>1</sup>, publicó la *Norma Internacional ISO/IEC 9126*, la cual establece los lineamientos generales para la evaluación del producto de software a partir de seis categorías de calidad: funcionalidad, fiabilidad, *eficiencia*, usabilidad, *portabilidad* y *mantenibilidad*. (Figura 1).

---

<sup>1</sup> The International Organization for Standardization [www.iso.org/](http://www.iso.org/)

Figura 1: Norma internacional ISO/IEC 9126



Fuente: ISO/IEC 9126-1 (1991). JTC 1/SC 7. Information technology - Software product evaluation - Quality characteristics and guidelines for their use, JTC 1 Organization, Montreal, Quebec

Como se ilustra en la figura, cada una de las seis categorías esta determinada por subcategorías que la concretan, y éstas a su vez, por indicadores que permiten detectar las bondades del producto software. A continuación se detalla esta categorización.

**Categoría Funcionalidad:** subcategorizada por:

**Adecuación:** Capacidad del producto software para proporcionar un conjunto apropiado de funciones para tareas y objetivos de los usuario especificados.

**Exactitud:** Capacidad del producto software para proporcionar los resultados o efectos correctos o acordados, con el grado necesario de precisión.

**Interoperatividad:** Capacidad del producto software para interactuar con uno o más sistemas especificados.

**Seguridad de acceso:** Capacidad del producto software para proteger información y datos de manera que las personas o sistemas no autorizados no puedan leerlos o modificarlos, al tiempo que no se deniega el acceso a las personas o sistemas autorizados

**Cumplimiento funcional:** Capacidad del producto software para adherirse a normas, convenciones o regulaciones en leyes y prescripciones similares relacionadas con funcionalidad.

**Categoría Fiabilidad:** subcategorizada por:

**Madurez:** Capacidad del producto software para evitar fallar como resultado de fallos en el software.

**Tolerancia a fallos:** Capacidad del software para mantener un nivel especificado de prestaciones en caso de fallos del software o de infringir sus interfaces especificadas.

**Recuperabilidad:** Capacidad del producto software para reestablecer un nivel de prestaciones especificado y de recuperar los datos directamente afectados en caso de fallo.

**Cumplimiento de la fiabilidad:** Capacidad del producto software para adherirse a normas, convenciones o regulaciones relacionadas con la fiabilidad.

**Categoría Usabilidad:** subcategorizada por:

**Entendimiento:** Capacidad del producto software que permite al usuario entender si el software es adecuado y cómo puede ser usado para unas tareas o condiciones de uso particulares.

**Aprendizaje:** Capacidad del producto software que permite al usuario aprender sobre su aplicación.

**Operabilidad:** Capacidad del producto software que permite al usuario operarlo y controlarlo.

**Atracción:** Capacidad del producto software para ser atractivo al usuario.

**Cumplimiento de la usabilidad:** Capacidad del producto software para adherirse a normas, convenciones, guías de estilo o regulaciones relacionadas con la usabilidad.

**Categoría Eficiencia:** subcategorizada por:

**Comportamiento temporal:** Capacidad del producto software para proporcionar tiempos de respuesta, tiempos de proceso y potencia apropiados, bajo condiciones determinadas.

**Utilización de recursos:** Capacidad del producto software para usar las cantidades y tipos de recursos adecuados cuando el software lleva a cabo su función bajo condiciones determinadas.

**Cumplimiento de la eficiencia:** Capacidad del producto software para adherirse a normas o convenciones relacionadas con la eficiencia.

**Categoría Mantenibilidad:** subcategorizada por:

**Capacidad para ser analizado:** Es la capacidad del producto software para serle diagnosticadas deficiencias o causas de los fallos en el software, o para identificar las partes que han de ser modificadas.

**Cambiabilidad:** Capacidad del producto software que permite que una determinada modificación sea implementada.

**Estabilidad:** Capacidad del producto software para evitar efectos inesperados debidos a modificaciones del software.

**Capacidad para ser probado:** Capacidad del producto software que permite que el software modificado sea validado.

**Cumplimiento de la mantenibilidad:** Capacidad del producto software para adherirse a normas o convenciones relacionadas con la mantenibilidad.

**Categoría Portabilidad:** subcategorizada por:

**Adaptabilidad:** Capacidad del producto software para ser adaptado a diferentes entornos especificados, sin aplicar acciones o mecanismos distintos de aquellos proporcionados para este propósito por el propio software considerado.

**Instalabilidad:** Capacidad del producto software para ser instalado en un entorno especificado.

**Coexistencia:** Capacidad del producto software para coexistir con otro software independiente, en un entorno común, compartiendo recursos comunes.

**Reemplazabilidad:** Capacidad del producto software para ser usado en lugar de otro producto software, para el mismo propósito, en el mismo entorno.

**Cumplimiento de la portabilidad:** Capacidad del producto software para adherirse a normas o convenciones relacionadas con la portabilidad.

### **PARÁMETROS DE LA PROPUESTA DE EVALUACIÓN DEL SOFTWARE EDUCATIVO.**

Nuestro papel como docentes y usuarios de recursos y materiales instruccionales que se concretan en distintos programas y software educativo, nos impone la necesidad de elegir productos “efectivos y de calidad”, que contribuyan al desarrollo del proceso de enseñanza aprendizaje. Bajo esta perspectiva, cobra relevancia la valoración del software desde parámetros que ofrezcan información sobre su efectividad.

Así, se propone valorar software educativo tomando como referentes tres de las seis categorías que sobresalen en el modelo ISO/IEC 9126. En este sentido, se estima que la base de los parámetros de evaluación del software educativo, debe estar integrada por:

- **La Funcionalidad** ya que es una característica fundamental en todo software; ella determina la capacidad del producto del software para proveer funciones que cumplan con necesidades específicas o implícitas, cuando el software es utilizado bajo ciertas condiciones.
- **La Usabilidad** debido a que, para que un software educativo motive al aprendizaje, es fundamental que el material educativo sea atractivo y de fácil manejo; debe generar actividades interactivas que motiven y mantengan la atención, actividades variadas y que respondan a los diversos estilos de aprendizaje. Esta categoría se refiere a la capacidad del producto de software para ser atractivo, entendido, aprendido y utilizado por el usuario bajo condiciones específicas.

- **La Fiabilidad** debido a que es importante que el producto funcione bajo las condiciones establecidas y mantenga un nivel específico de rendimiento para garantizar un ambiente de aprendizaje adecuado bajo condiciones especificadas.

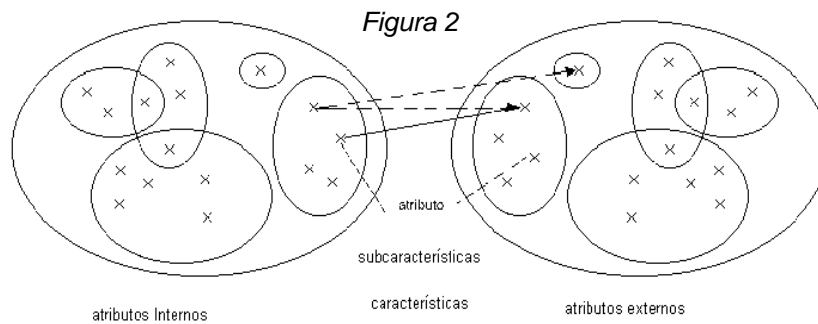
Con la intención de sistematizar la propuesta, en las siguientes líneas se ofrece la subcategorización<sup>1</sup> y la asignación de las métricas correspondientes a cada una de las categorías en referencia; estos aportes permitirán, finalmente, arrojar los resultados de eficiencia o no del software educativo.

Para la categoría *funcionalidad*, las subcategorías base: adecuación al propósito, precisión y seguridad

Para la categoría *usabilidad*, las subcategorías base: facilidad de comprensión, capacidad de uso, interfaz gráfica y operabilidad

Para la categoría *fiabilidad*, las subcategorías base: sentido, desempeño y tolerancia a fallas.

Es importante destacar que para la definición de las categorías del modelo, ha sido considerada la relación de influencia entre atributos internos y externos del software. Así, los niveles de ciertos atributos internos se han encontrado para influir en los niveles de algunos atributos externos (Figura 2), de modo que haya un aspecto externo y un aspecto interno en la mayoría de las categorías. Por ejemplo, la fiabilidad puede ser medida externamente observando el número de fallas en un período dado del tiempo de ejecución durante un ensayo del software, e internamente examinando las especificaciones detalladas y el código fuente para determinar el nivel de la tolerancia de falla. De esta manera los atributos internos serán los indicadores de los atributos externos.



Fuente: ISO/IEC 9126-1 (1991). JTC 1/SC 7. Information technology - Software product evaluation-Quality characteristics and guidelines for their use, JTC 1 Organization, Montreal, Quebec

1 \* Denominación pertinente al proceso de categorización propio de la investigación cualitativa. Las subcategorías en evaluación de software se denominan características , y éstas pueden estar divididas a su vez, en subcaracterísticas

Un atributo interno puede influenciar a una o más subcategorías, y una categoría puede estar influenciada por más de un atributo. En este modelo la totalidad de atributos de la calidad del producto de software, se clasifica en una estructura arborescente jerárquica de categorías y de subcategorías. El nivel más alto de esta estructura consiste en *categorías de calidad* y el nivel más bajo consiste en *atributos de calidad de software*. La jerarquía no es perfecta cuando algunos atributos pueden contribuir a más de una subcategoría.

Finalmente, es necesario asignar *métricas* para cada subcategoría, que permitan operacionalizar la propuesta del modelo al área de software educativo.

Según la Oficina Nacional de Gobierno Electrónico e Informática del Perú<sup>1</sup>, dependencia creadora de las normas métricas del software a ser utilizado en el sector público de dicho país, las métricas deben derivarse o ponderarse bajo los siguientes parámetros:

1. El área de informática aplicará el tipo de escala de proporción.
2. A cada métrica seleccionada le asignará un puntaje máximo de referencia.
3. La suma de los puntajes máximos de todas las métricas deberá ser igual a 100 puntos.
4. El área de informática podrá establecer niveles de calificación cualitativa en base a los puntajes como por ejemplo:
  - Puntaje mínimo de aprobación.
  - Inaceptable, mínimo aceptable, rango objeto, excede los requisitos.
  - Insatisfactorio, satisfactorio.
  - Se pueden usar números hasta con un decimal de aproximación. (Ejemplos: 4.1, 3.8, 11.7).
5. El área de informática podrá establecer, por cada métrica, un puntaje mínimo de aprobación. En caso de que no se alcance ese puntaje, se considerará que el producto de software no cumple con las necesidades de información y debe ser rechazado.

A continuación se presenta un ejemplo de *normalización de los resultados* de las métricas a una escala del 1 al 5.

---

<sup>1</sup> <http://www.ongei.gob.pe/>

Normalización de resultados

No aplica o No sabe	1	2	3	4	5
	Nunca	Pocas veces	Algunas veces	Casi siempre	Siempre
	Muy mal	Mal	Normal	Bueno	Muy bueno
	Nada	Alguno	Normal	Casi	Mucho
	No tiene	Básico	Mediano	Alto	Muy alto
No aplica (NA)	Ninguno	Muy pocos	Pocos	Casi todos	Todos
	Muy baja	Baja	Media	Alta	Muy alta
No sabe (NS)	No	Poco	Medianamente	Casi todo	Completamente
	Inaceptable	Debajo del promedio	promedio	Buena	Excelente
	No está definido	Poco definido	Medianamente definido	Casi todo definido	Completamente definido
	No		Promedio		Si

Fuente: LISI - USB.

Según el LISI-USB se debe verificar, de acuerdo al ejemplo, que el 75% de las métricas se encuentren dentro de los valores óptimos (mayor o igual a 4) para cada una de sus subcaracterísticas (subcategorías) y características (categorías). Si no se cumple el 75% de las métricas asociadas, entonces esta subcaracterística o característica tendrá calidad nula.

Respecto a los evaluadores, en el caso de institutos educativos, serán los docentes de aula, los de informática y los estudiantes los más indicados para ello; y en el caso de empresas desarrolladoras de software educativo, la población de evaluadores serán los expertos en contenido, diseño instruccional, en informática, entre otros.

Se debe tener en cuenta que en toda fase de *preparación de la evaluación* se especifican tres pasos generales, a saber:

1. Selección de las Métricas de Calidad
2. Definición de los Niveles de Puntaje y
3. Definición de los Criterios de Valoración.

En el primer paso (o actividad) los evaluadores deben decidir y seleccionar las métricas directas (o indirectas), debido a que las características definidas en el estándar, son de muy alto nivel. Las métricas seleccionadas, las cuales deben correlacionarse con las características respectivas, difieren del contexto y de la fase del proceso de desarrollo.

En el segundo paso, (*Definición de los Niveles de Puntaje*), se deben definir los niveles de satisfacción. Esto es necesario debido a que el valor de una métrica, es calculado en un rango de una escala, que no expresa el nivel de

satisfacción. Por lo tanto, el estándar indica que se deben definir grados o rangos de valores de satisfacción de los requerimientos. El estándar indica que, debido a que la calidad se refiere a necesidades dadas y específicas, ningún nivel de puntaje general es posible sino que se debe definir con respecto a cada evaluación en particular.

En el tercer paso (*Definición de los Criterios de Valoración*) de la fase *Preparación de la Evaluación*, los evaluadores deben definir procedimientos para resumir los resultados de las diferentes características. Se pueden usar tablas de decisión o promedios pesados. El procedimiento puede incluir otros aspectos como tiempo y costos, en un contexto particular.

### **Apreciación final**

Debemos valorar que la incorporación de la Informática, la computación y la Internet a través de las Tic's ofrecen distintas dimensiones al proceso de enseñanza aprendizaje. Garantizar que esta incorporación sea de calidad no es una tarea fácil, debido al carácter multidimensional de la calidad. Se espera que partiendo del estándar ISO 9126 se contribuya en la creación de un modelo que permita evaluar el software educativo y derive en una verdadera innovación que fortalezca la calidad de la infoeducación.

### **REFERENCIAS BIBLIOGRÁFICAS**

- Callaos, N. y Callaos, B. (1993). *Designing with Systemic Total Quality*, en International Conference on Information Systems, International Institute of Informatics and Systemics, Orlando.
- Gros, B. (2000). *Del software educativo a educar con software*. Artículo publicado en el número monográfico de la revista digital Quaderns Digital. En <http://www.quadernsdigitals.net/articuloquaderns.asp?IdArticle=3743>
- Grimán, A, Mendoza, L., Pérez M., Rojas, T. (2001). *Modelo de calidad de software educativo: aplicación del estándar ISO/IEC 9126*. Valencia: 2da. Edición EDUWEB – Congreso Venezolano de Educación e Informática. Julio. 2001, Venezuela.
- ISO/IEC 9126-1 (1991). JTC 1/SC 7. *Information technology - Software product evaluation - Quality characteristics and guidelines for their use*, JTC 1 Organization, Montreal, Quebec.
- Publicaciones del Laboratorio de investigación en sistemas de Información de la Universidad Simón Bolívar. En:<http://www.lisi.usb.ve/publicaciones>
- Pressman, R. (2002). *Ingeniería de Software: Un enfoque Práctico*. 5ta. Edición. McGraw Hill.