

# Calidad en la Industria del Software. La Norma ISO-9126

María Antonieta Abud Figueroa\*

## Introducción

**H**oy en día las compañías de todo el mundo industrializado reconocen que la calidad del producto se traduce en ahorro de costos y en una mejora general. La industria de desarrollo de software no es la excepción, por lo que en los últimos años se han realizado intensos trabajos para aplicar los conceptos de calidad en el ámbito del software.

Hablar de calidad del software implica la necesidad de contar con parámetros que permitan establecer los niveles mínimos que un producto de este tipo debe alcanzar para que se considere de calidad. El problema es que la mayoría de las características que definen al software no se pueden cuantificar fácilmente; generalmente, se establecen de forma cualitativa, lo que dificulta su medición, ya que se requiere establecer métricas que permitan evaluar cuantitativamente cada característica dependiendo del tipo de software que se pretende calificar.

En este sentido se han realizado muchos trabajos que establecen propuestas para el establecimiento de los factores cualitativos que afectan la calidad del software. Entre los principales están los factores de calidad de McCall [1][4] y aquellos propuestos por Hewlett-Packard (FURPS: Funcionalidad,

Usability, Reliability; Performance, Supportability) [4].

Además se han hecho varios intentos por estandarizar los mecanismos de evaluación de calidad del software. Entre los principales están la familia de normas ISO 9000 (en especial la ISO 9001 y la ISO 9003-2)[5], el modelo de niveles madurez CMM (Capability Maturity Model)[7], el estándar para el aseguramiento de planes de calidad del IEEE 730:1984 [7], el plan general de garantía de calidad del Consejo Superior de Informática MAP[7] y la norma ISO/IEC 9126 [3], que es objeto de este estudio.

En este trabajo se expone un esquema general del estándar ISO 9126, con el fin de mostrar los elementos que deben considerarse en la evaluación de calidad de los productos de software de acuerdo a este estándar, de modo que todo aquél que se interese

en aplicar modelos de calidad en la producción de software pueda generar sus propias métricas bajo la guía de este estándar.

## Modelo de Calidad Establecido por el estándar ISO 9126

La ISO, bajo la norma ISO-9126, ha establecido un estándar internacional para la evaluación de la calidad de productos de software el cual fue publicado en 1992 con el nombre de "Information technology -Software product evaluation: Quality

characteristics and guidelines for their use", en el cual se establecen las características de calidad para productos de software.

El estándar ISO-9126[7] establece que cualquier componente de la calidad del software puede ser descrito en términos de una o más de seis características básicas, las cuales son: funcionalidad, confiabilidad, usabilidad, eficiencia, mantenibilidad y portabilidad; cada una de las cuales se detalla a través de un conjunto de subcaracterísticas que permiten profundizar en la evaluación de la calidad de productos de software. La tabla 1 muestra la pregunta central que atiende cada una de estas características.

Características	Pregunta central
Funcionalidad	¿Las funciones y propiedades satisfacen las necesidades explícitas e implícitas; esto es, el qué . . . ?
Confiabilidad	¿Puede mantener el nivel de rendimiento, bajo ciertas condiciones y por cierto tiempo?
Usabilidad	¿El software es fácil de usar y de aprender?
Eficiencia	¿Es rápido y minimalista en cuanto al uso de recursos?
Mantenibilidad	¿Es fácil de modificar y verificar?
Portabilidad	¿Es fácil de transferir de un ambiente a otro?

Tabla 1. Características de ISO-9126 y aspecto que atiende cada una.

## Características Propuestas por ISO-9126

A continuación se detalla cada una de las características que establece el estándar ISO-9126.

### C1. Funcionalidad

En este grupo se conjunta una serie de atributos que permiten calificar si un producto de software maneja en forma adecuada el conjunto de funciones que satisfagan las necesidades para las cuales fue diseñado. Para este propósito se establecen los siguientes atributos:

\*Maestra en Sistemas de Información por el Instituto Tecnológico y de Estudios Superiores de Monterrey-Campus Morelos. Profesor-Investigador en la División de Estudios de Posgrado e Investigación del Instituto Tecnológico de Orizaba. Correo electrónico: mabud@itorizaba.edu.mx.

- **Adecuación.** Se enfoca a evaluar si el software cuenta con un conjunto de funciones apropiadas para efectuar las tareas que fueron especificadas en su definición.
- **Exactitud.** Este atributo permite evaluar si el software presenta resultados o efectos acordes a las necesidades para las cuales fue creado.
- **Interoperabilidad.** Permite evaluar la habilidad del software de interactuar con otros sistemas previamente especificados.
- **Conformidad.** Evalúa si el software se adhiere a estándares, convenciones o regulaciones en leyes y prescripciones similares.
- **Seguridad.** Se refiere a la habilidad de prevenir el acceso no autorizado, ya sea accidental o premeditado, a los programas y datos.

## ***C2. Confiabilidad***

Aquí se agrupan un conjunto de atributos que se refieren a la capacidad del software de mantener su nivel de ejecución bajo condiciones normales en un periodo de tiempo establecido. Las subcaracterísticas que el estándar sugiere son:

- **Nivel de Madurez.** Permite medir la frecuencia de falla por errores en el software.
- **Tolerancia a fallas.** Se refiere a la habilidad de mantener un nivel específico de funcionamiento en caso de fallas del software o de cometer infracciones de su interfaz específica.
- **Recuperación.** Se refiere a la capacidad de restablecer el nivel de operación y recobrar los datos que hayan sido afectados directamente por una falla, así como al tiempo y el esfuerzo necesarios para lograrlo.

## ***C3. Usabilidad***

Consiste de un conjunto de atributos que permiten evaluar el esfuerzo necesario que deberá invertir el usuario para utilizar el sistema.

- **Comprensibilidad.** Se refiere al esfuerzo requerido por los usuarios para reconocer la estructura lógica del sistema y los conceptos relativos a la aplicación del software.
- **Facilidad de Aprender.** Establece atributos del software relativos al esfuerzo que los usuarios deben hacer para aprender a usar la aplicación.
- **Operabilidad.** Agrupa los conceptos que evalúan la operación y el control del sistema.

## ***C4. Eficiencia***

Esta característica permite evaluar la relación entre el nivel de funcionamiento del software y la cantidad de recursos usados. Los aspectos a evaluar son:

- **Comportamiento con respecto al Tiempo.** Atributos del software relativos a los tiempos de respuesta y de procesamiento de los datos.
- **Comportamiento con respecto a Recursos.** Atributos del software relativos a la cantidad de recursos usados y la duración de su uso en la realización de sus funciones.

## ***C5. Mantenibilidad***

Se refiere a los atributos que permiten medir el esfuerzo necesario para realizar modificaciones al software, ya sea por la corrección de errores o por el incremento de funcionalidad. En este caso, se tienen los siguientes factores:

- **Capacidad de análisis.** Relativo al esfuerzo necesario para diagnosticar las deficiencias o causas de fallas, o para identificar las partes que deberán ser modificadas.
- **Capacidad de modificación.** Mide el esfuerzo necesario para modificar aspectos del software, remover fallas o adaptar el software para que funcione en un ambiente diferente.
- **Estabilidad.** Permite evaluar los riesgos de efectos inesperados

debidos a las modificaciones realizadas al software.

- **Facilidad de Prueba.** Se refiere al esfuerzo necesario para validar el software una vez que fue modificado.

## ***C6. Portatibilidad***

En este caso, se refiere a la habilidad del software de ser transferido de un ambiente a otro, y considera los siguientes aspectos:

- **Adaptabilidad.** Evalúa la oportunidad para adaptar el software a diferentes ambientes sin necesidad de aplicarle modificaciones.
- **Facilidad de Instalación.** Es el esfuerzo necesario para instalar el software en un ambiente determinado.
- **Conformidad.** Permite evaluar si el software se adhiere a estándares o convenciones relativas a portatibilidad.
- **Capacidad de reemplazo.** Se refiere a la oportunidad y el esfuerzo usado en sustituir el software por otro producto con funciones similares.


## ***Conclusiones***

El mundo globalizado exige cada vez más la aplicación de estándares internacionales que garanticen la calidad de los productos. Por esta razón, es necesario que todo aquel que se dedica al desarrollo de software incluya en sus procesos, estándares de calidad que permitan certificarse en alguno de los modelos.

Aquí se ha presentado un estándar, el ISO-9126, el cual establece una guía para la evaluación de la calidad del software, sin embargo es necesario que cada empresa dedicada a producir software trabaje en establecer su modelo de calidad que le permita valorar el nivel de excelencia de sus productos, en el que deberán incluirse instrumentos de medición que permitan calificar cuantitativamente cada una de las características aquí

presentadas. Es importante mencionar, que dependiendo de los distintos tipos de aplicaciones las

métricas podrán variar, ya que aunque las características expuestas son comunes a la totalidad de

los productos, cada software particular requiere una evaluación específica. 

## **Bibliografía**

- [1] Cervera Paz, Ángel. *El modelo de McCall como aplicación de la calidad a la revisión del software de gestión empresarial*. Universidad de Cádiz, obtenido el 24 de abril del 2001 del sitio web <http://www.monografias.com/trabajos5/call/call.html>
- [2] Fairley, Richard. *Ingeniería de Software*, 2ª Edición. Editorial Mc Graw Hill. México. 1987.
- [3] International Organization for Standardization. *Norma de gestión de la calidad y garantía de la calidad, parte 3*. Sitio web <http://alarcos.inf-cr.uclm.es/doc/calidad/ISO%209000-3.doc>.
- [4] Pressman, Roger S. *Ingeniería de Software, Un enfoque práctico*, 4ª. Edición. Editorial Mc Graw Hill. México. 1998.
- [5] Quintanilla Osorio, Gloria. “La implantación de ISO 9001 en el desarrollo de software”, *Revista Soluciones Avanzadas*, Septiembre (1999), p.31.
- [6] Rodríguez G., González J., Dávila Gladys. “La norma ISO 9001 en una fábrica de software a la medida”, *Revista Soluciones Avanzadas*, julio (1998), p.27.
- [7] Sanders, Joc & Eugene Curran. *Software Quality. A Framework for Success in Software Development and Support*, Addison Wesley.

