

## Métricas estadísticas asociadas al proceso de desarrollo y mejora continua de software

**Ana María Strub**

Universidad Tecnológica Nacional  
[anastrub@gmail.com](mailto:anastrub@gmail.com)

**Cecilia Andrea Savi**

Universidad Tecnológica Nacional  
[ceciliasavi@hotmail.com](mailto:ceciliasavi@hotmail.com)

**Mariel Ferrando**

Universidad Tecnológica Nacional  
[mariel\\_ferrando@hotmail.com](mailto:mariel_ferrando@hotmail.com)

**Irene Romoli**

Universidad Tecnológica Nacional  
[ire.romoli@gmail.com](mailto:ire.romoli@gmail.com)

**Clarisa Stefanich**

Universidad Tecnológica Nacional  
[clarystefanich@gmail.com](mailto:clarystefanich@gmail.com)

### Resumen

Los productos de software se han convertido en uno de los principales objetivos estratégicos de las organizaciones debido a que, los procesos más importantes de estas, dependen del buen funcionamiento de los sistemas de software.

La gran mayoría de los estudios de calidad de software, están enfocados a las actividades de administración de los proyectos de desarrollo de software y solo en algunos entornos industriales y académicos, la calidad del software ha sido evaluada mediante estudios analíticos. En este aspecto se está evolucionado hacia modelos formales estadísticos, que se basan en métricas apropiadas para el control y evaluación de la calidad de un producto de software.

Este proyecto involucra un análisis del estado del arte en ese contexto, que incluye un estudio comparado sobre los modelos de calidad vigentes y sus niveles de evaluación

para productos relacionados con la Ingeniería del Software. Identificar los atributos y descubrir las métricas de evaluación, priorizando aquellas que utilicen componentes de probabilidad y estadística para su cálculo. El objetivo final está dirigido a detallar un protocolo que oriente el análisis y la evaluación de la calidad en los productos de software. Como transferencia pretendemos que esta metodología sirva de modelo para cualquier organización-empresa que requiera una evaluación de calidad en sus productos de software.

**Palabras clave:** Calidad software. Métricas estadísticas.

---

## Introducción

La gestión de la calidad se puede entender como el conjunto de actividades y medios necesarios para definir e implantar un sistema de la calidad, por una parte, y responsabilizarse de su control, aseguramiento y mejora continua, por otra. En este sentido, la gestión de la calidad en cualquier organización y sobre todo en las dedicadas al desarrollo y mantenimiento de software, se ocupan a nivel de proyecto, de las distintas actividades de desarrollo análisis de productos y mantenimiento de software que deben ser adaptadas a las características concretas del proyecto y de su entorno, para ser aplicadas en la práctica.

La gestión de la calidad ha seguido dos líneas que pueden ser complementarias entre sí: Por una parte, se ha seguido la línea marcada por las entidades internacionales de estandarización para todas las organizaciones de producción o servicios, principalmente, se ha impuesto en la práctica las directrices marcadas por ISO (Organization for International Standardization) a través de su serie de normas ISO 9000 para la gestión de calidad. En el caso del software es principalmente aplicable la norma ISO 9001 [ISO, 1994a], aunque en los últimos años se está incrementando el número de organizaciones de este sector adheridas a la norma ISO 9002 [ISO, 1994b].

Por otra parte, el mundo del software ha creado su propia línea de trabajo en la gestión de la calidad del software, es decir, trabajar sobre los procesos de producción de software como medio de asegurar la calidad del producto software. Así, se

comenzó en el SEI (Software Engineering Institute) de EE.UU. proponiendo un modelo de clasificación y mejora de los procesos empleados por las organizaciones de software denominado CMM [Paulk et al., 1993]. Su trabajo se centra en el estudio y clasificación de los distintos procesos involucrados en la producción de software bajo el enfoque de una serie de niveles de madurez. Sobre este modelo pionero, se han creado nuevos modelos que suponen tanto actualizaciones como variantes.

Dentro de este nivel de actuación, la medición supone, junto a las actividades de verificación y validación (básicamente, pruebas de software y actividades de revisión y auditoría), una de las técnicas principales previstas en los estándares para el control y el aseguramiento de la calidad [IEEE 1074, 1991].

Desde este punto de vista, la medición puede contribuir tanto en el control de los procesos y actividades como en el de los productos, lo que promueve dos importantes líneas de trabajo sobre medición: Por una parte, el concepto de calidad es demasiado complejo como para poder ser evaluado o medido mediante una única medida. La norma UNE-EN ISO 8402 [AENOR, 1995] define el aseguramiento de la calidad (AC) como "el conjunto de acciones planificadas y sistemáticas implantadas dentro del sistema de calidad, y demostrables si es necesario, para proporcionar la confianza adecuada de que una entidad cumplirá los requisitos para la calidad". Uno de los campos en los que más se ha trabajado es en la utilización de modelos de evaluación de calidad de software que tratan de aportar un medio para definir y descomponer el concepto de calidad de software en características más sencillas de evaluar y medir. Así, podemos encontrar modelos de evaluación generales como el FCM (Factores/Criterios/Métricas) [McCall et al., 1977], métodos para crear modelos propios en cada proyecto como COQUAMO [Kitchenham y Walker, 1989] o el de Gilb [Gilb, 1987] e, incluso, estándares que abordan esta cuestión como ISO 9126 [ISO 9126, 1991] o IEEE 1061 [IEEE 1061, 1992]. Por otra parte se siguen proponiendo mediciones, principalmente de productos, que proclaman su utilidad para evaluar la calidad del software, o alguna de sus facetas.

En el área de los Sistemas informáticos, está claro que en cada proyecto de desarrollo, el aseguramiento de la calidad del software supone la aplicación de guías de proceso o protocolos marcados por las disposiciones que se han establecido, bien sea como un sistema de calidad bien definido o bien mediante una serie de procedimientos y estándares preceptivos.

Los problemas principales a los que intentamos dar respuesta son: ¿Qué características reúnen los modelos de evaluación de software en el ámbito de la Ingeniería del software? ¿Cómo se calculan las métricas/medidas y cuáles son los procedimientos estadísticos involucrados? ¿Que aporte puede significar la definición de un protocolo de procedimientos, respecto a métricas/medidas vinculadas a la calidad de un software en la formación por competencias profesionales del Ingeniero en Sistemas?

Para y por ello el objetivo de nuestra investigación establece:

Definir un protocolo de procedimientos de actividades, cálculos, interpretaciones lo más estándar posible, que soporte la colección de datos, el análisis y las estimaciones de métricas asociadas a la mejora continua de software, en el ámbito de la Ingeniería del Software

Los objetivos específicos:

- \* Conocer y analizar distintos modelos de evaluación de calidad de software, con el fin de tener como referente, un modelo, el más apropiado a nuestra investigación.
- \* Identificar y clasificar los atributos y sus correspondientes medidas/métricas, vinculadas a la evaluación de la calidad de un producto de software.
- \* Determinar los procedimientos, operaciones e interpretaciones de todas las medidas que se utilizan en un proceso evaluativo, incluyendo especialmente aquellas medidas que presentan componentes de probabilidad y estadística.
- \* Enunciar y caracterizar el protocolo explicitando los procedimientos y cálculos que guían la obtención de las métricas.

## Contenido

A lo largo de la historia el término **calidad** ha sufrido numerosos cambios que conviene reflejar en cuanto su evolución. Son diferentes e innumerables las acepciones del término calidad de software, lo que nos obliga a posicionarnos en alguno de ellos, el más apropiado a nuestro campo de acción, Sistemas de Información.

A modo de síntesis, recuperamos definiciones de calidad, más próximas a nuestros objetivos, referenciando a diferentes autores y organizaciones, todos reconocidos en el campo de estudio a tratar.

### DRAE: Diccionario de la Real Académica Española

Propiedad o conjunto de propiedades inherentes a un objeto que permiten apreciarlo como mejor, igual o peor que otros objetos de su especie.

### ISO 8042:1994

Conjunto de propiedades y de características de un producto o servicio que le confieren capacidad para satisfacer necesidades expresadas o implícitas.

### Joseph M. Juran

Es la totalidad de los rasgos y características de un producto o servicio que se sustenta en su habilidad para satisfacer las necesidades establecidas implícitas.

### Armand V. Feigenbaum

La composición total de las características de los productos y servicios de marketing, ingeniería, fabricación y mantenimiento, a través de los cuales los productos y los servicios cumplirán las expectativas de los clientes

### Robert Pirsiq.

Calidad no es ni materia ni espíritu, sino una tercera entidad independiente de las otras dos, aun cuando la calidad no pueda definirse, usted sabe bien qué es.

### Harold L. Gilmore.

Calidad es la medida en que un producto específico se ajusta a un diseño o especificación.

### Kaoru Ishikawa

Calidad, significa desarrollar, diseñar y producir y mantener un producto que sea el más económico, el más útil y siempre satisfactorio para el consumidor.

William Edwards Deming

Calidad, es la aplicación de los principios y técnicas estadísticas en todas las fases de la producción, dirigida a la fabricación más económica de un producto (servicio) que es útil en grado máximo y que tiene mercado.

Cada una de estas definiciones tiene una visión desde perspectivas diferentes. La calidad desde:

*Una perspectiva de producto:* La calidad es diferenciarse cualitativa y cuantitativamente respecto de algún atributo requerido.

*Una perspectiva de usuario:* La calidad implica la capacidad de satisfacer los deseos de los consumidores. La calidad de un producto depende de cómo éste responda a las preferencias de los clientes, por lo que se dice que la calidad es adecuación al uso.

*Una perspectiva de producción:* La calidad puede definirse como la conformidad relativa con las especificaciones, a lo que al grado en que un producto cumple las especificaciones del diseño, entre otras cosas, mejora su calidad.

*Una perspectiva de valor:* La calidad significa aportar valor al cliente, esto es, ofrecer unas condiciones de uso del producto o servicio superiores a las que el cliente espera recibir y a un precio accesible.

El término “calidad”, como hemos analizado hasta el momento, tiene innumerables acepciones. En nuestro caso vamos a posicionarnos en que la calidad debe ser mensurable y predecible, y debe incorporar factores básicos como: ausencia de defectos, satisfacción del usuario y conformidad con los requerimientos.

En el caso particular de las organizaciones orientadas al desarrollo de software, las mediciones permiten mejorar sus procesos, ayudar en la planificación, darle seguimiento a los mismos y controlar determinados proyectos, en aras de lograr una mejor calidad. “**Calidad de Software**”, en el marco del área Ingeniería de software implica aspectos como:

Concordancia con los requisitos funcionales debidamente establecidos, con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente. Los requisitos de software son la base de la medida de calidad. Pressman, 1992.

---

La calidad del software es el grado con el que un sistema, componente o proceso cumple los requerimientos especificados y las necesidades o expectativas del cliente o usuario. IEEE, Std 610-1900.

---

Calidad del software: es el desarrollo de software basado en estándares con la funcionalidad y rendimiento total que satisfacen los requerimientos del cliente.

Procesos de desarrollo, artefactos, gestión de proyectos, análisis y diseño, especificación de requerimientos, arquitectura, son solo algunos de los componentes que se aglomeran para conformar la ingeniería de software (IS) como disciplina para la creación y mantenimiento de software. Dentro de ésta, existe un subconjunto de teorías, herramientas y métodos orientados a lo que se denomina la calidad del software.

Sitio web: <http://www.mitecnologico.com/Main/DefinicionCalidadDeSoftware>

---

La calidad del software es el conjunto de cualidades que lo caracterizan y que determinan su utilidad y existencia. La calidad es sinónimo de eficiencia, flexibilidad, corrección, confiabilidad, Mantenibilidad, portabilidad, Usabilidad, seguridad e integridad. La calidad del software es medible y varía de un sistema a otro o de un programa a otro. La calidad del software puede medirse después de elaborado el producto. Pero esto puede resultar muy costoso si se detectan problemas derivados de imperfecciones en el diseño, por lo que es imprescindible tener en cuenta tanto la obtención de la calidad como su control durante todas las etapas del ciclo de vida del software.

Sitio Web: [http://www.bvs.sld.cu/revistas/aci/vol3\\_3\\_95/aci05395.htm](http://www.bvs.sld.cu/revistas/aci/vol3_3_95/aci05395.htm)

---

Pressman modifica su definición en algunos aspectos y define calidad como: Concordancia del software producido con los requerimientos explícitamente establecidos, con los estándares de desarrollo prefijados y con los requerimientos implícitos no establecidos formalmente, que desea el usuario. Pressman, 1998

---

Buscando mayor especificidad en la investigación abordada y dentro de un marco general de modelos existentes y orientados a la mejora de procesos, productos y proyectos con sus características generales, nos centramos en aquellos que se preocupan y ocupan en certificar calidad, y sus niveles de evaluación específicamente de productos relacionados con la Ingeniería del Software.

**Un modelo** es un marco conceptual que especifica una serie de conceptos medibles y sus relaciones, para una determinada categoría de entidad. Es decir, un modelo (de calidad) evalúa uno o varios conceptos medibles.

Dentro del conjunto de modelos y metodologías de mejora de la calidad, existen un conjunto de modelos que sitúan el foco de la mejora no en el proceso, sino en el propio producto final. Estos modelos son denominados Modelos de Evaluación de Calidad de Productos de Software. No obstante, y a pesar de que los pilares básicos del desarrollo de software son comunes para todas las organizaciones, por regla general la aplicación de este tipo de aproximaciones a la calidad mediante los modelos orientados a producto dependen de cada organización, por lo que las soluciones no son fácilmente generalizables. Cualquiera de los modelos vigentes sugieren indicadores y mediciones que permiten a una organización analizar su desempeño, mejorar sus prácticas y consecuentemente entregar productos de software de alta calidad. Pero también cada organización elige la forma de definir dichos indicadores, por lo tanto, es difícil hablar de estandarización a nivel de mediciones. Actualmente a la mayor parte de las pequeñas organizaciones les resulta complicado reconocer qué es lo que vale la pena medir y cómo medirlo.

Construir un modelo de calidad es bastante complejo y es usual que estos modelos descompongan la calidad del producto de software jerárquicamente en una serie de

características y sub características que pueden usarse como una lista de comprobación de aspectos relacionados con la calidad. Se han desarrollado varios modelos de calidad para diferentes productos y procesos de software.

La mayor parte de ellos están basados en la norma ISO 9126:2001. Esta norma define un conjunto de características de calidad que son después refinadas en sub características que están descompuestas en atributos. Los valores de estos atributos se calculan mediante la utilización de métricas.

A modo referencial mencionamos algunos modelos actuales, que se basan en esta norma, el modelo Simão y Belchior (2003) en el que los autores han ampliado las sub características y atributos propuestos por la norma llegando a identificar 124 atributos de calidad para los componentes software.

El modelo de calidad QUINT2 (Niessink, 2002) también presenta una ampliación de la norma ISO 9126, pensada para valorar la calidad de arquitecturas software.

El modelo de calidad propuesto por Franch and Carvallo (2003) presenta una adaptación de la ISO9126 para correo electrónico.

Botella et al. (2003) proponen un modelo para la selección de ERP, seleccionan como marco de trabajo el estándar de calidad ISO/IEC 9126-1 y proponen una metodología para adaptarlo a su dominio específico

En Zo and Ramamurthy (2002) los autores presentan un modelo para valorar y seleccionar los sitios Web de comercio electrónico en un entorno B2C (Business-to-Consumer).

En Webb and Webb (2002) se presentan los factores de calidad del sitio Web que son importantes para los consumidores.

En Parasuraman et al (1998) se describe el modelo SERVQUAL el cual contiene 5 dimensiones y 22 ítems para medir los diferentes elementos de la calidad de un servicio en general.

En el caso de WQM (Web Quality Model) la gran presencia de tecnología Web y la gran información asociada a esta tecnología hace imprescindible que los diseños se realicen bajo unos mínimos criterios de calidad, hasta ahora prácticamente inexistentes.. WQM

está caracterizado por tres elementos: (1) La característica de calidad (basada en Quint2 y en la ISO 9126), (2) El proceso del ciclo de vida (basado en la ISO12207) y (3) Características (contenido, presentación y navegación).

PQM (Portal Quality Model) tiene como objetivo definir un modelo de calidad para portales, denominado PQM, para lo que se ha utilizado el método GQM. El modelo consta de 6 dimensiones: tangibles, confiabilidad, capacidad de respuesta, aseguramiento, empatía y calidad de los datos.

A partir de un relevamiento realizado a modelos y normas vigentes, y después de un exhaustivo análisis, se toman específicamente como referentes significativos para la investigación las Norma ISO/IEC-9126 y la ISO/IEC-14598, y la Norma ISO 25000. De la misma forma se escogen los modelos: MODELO DE MCCALL, MODELO DE DROMEY, como los más completos a nuestro criterio, al momento de evaluar la calidad de productos de software.

## McCall

---

Con su modelo de calidad, McCall intentó cerrar la brecha entre los usuarios y los desarrolladores enfocándose en una serie de factores de calidad de software que reflejara tanto el punto de vista de los usuarios como las prioridades de los desarrolladores.

El modelo presenta algunas características, que pueden resumirse a través de los siguientes puntos

- ✚ Se focaliza en el producto final, identificando atributos claves desde el punto de vista del usuario.
- ✚ Estos atributos se denominan factores de calidad y son normalmente atributos externos. Aunque también se incluyen algunos atributos que pueden ser internos.
- ✚ Los factores de calidad son demasiados abstractos para ser medidos, directamente, por lo que por cada uno de ellos se introducen atributos de bajo nivel denominados criterios de calidad.

✚ Algunos criterios de calidad son atributos internos, reflejando la creencia de McCall que el atributo interno tiene un efecto directo en el atributo externo correspondiente.

✚ Un nivel más de descomposición es necesario, mapeando cada criterio de calidad en un conjunto de métricas de calidad que son atributos (tanto del producto como del proceso) de muy bajo nivel, medibles directamente

El modelo de calidad de McCall tiene, tres perspectivas principales para definir e identificar la calidad de un producto de software:

1- La revisión de producto incluye: (habilidad de sufrir cambios).

- Mantenibilidad (el esfuerzo requerido para localizar y solucionar las fallas en un programa sin encontrarse en su ambiente operativo),
- Flexibilidad (la facilidad de hacer los cambios requeridos por los cambios en el entorno operativo) y
- Testeabilidad (la facilidad de las pruebas del programa, para asegurarse de que está libre de errores y cumple con su especificación).

2- La transición de producto está relacionado con: (adaptabilidad a un nuevo ambiente).

- Portabilidad (el esfuerzo requerido para transferir un programa desde un ambiente de trabajo a otro),
- Reusabilidad (la facilidad de reusar software en un contexto diferente) y la interoperabilidad (el esfuerzo requerido para acoplar un sistema a otro).

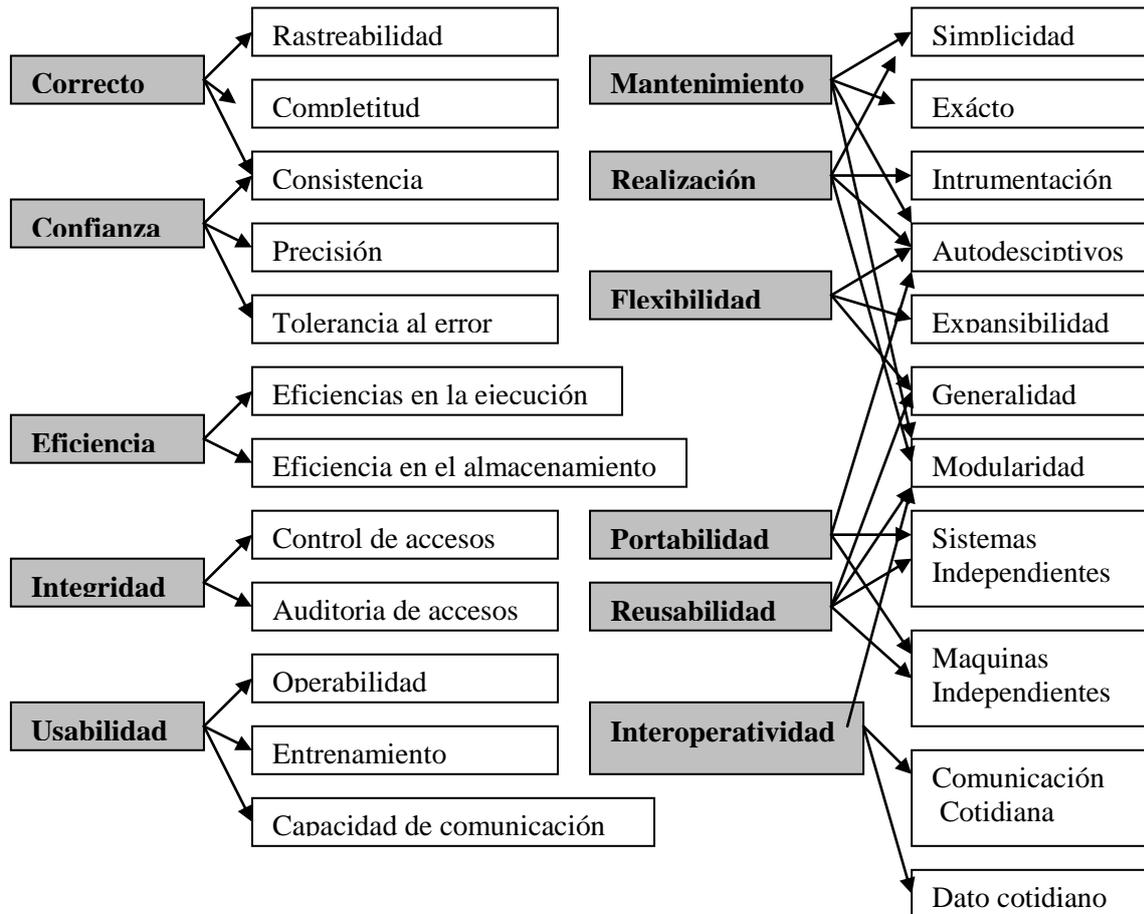
3- La calidad de las operaciones de productos dependen de: (características operativas).

- Corrección (la medida en la cual un programa cumple con las especificaciones),
- Fiabilidad (la habilidad del sistema para no fallar),

- Eficiencia (ya sea, categorizada en eficiencia en la ejecución o en el almacenamiento y en general, es decir, el uso de recursos, el tiempo de procesamiento, etc.),
- Integridad (la protección del programa al acceso no autorizado)
- Usabilidad (la facilidad de uso del software).

El modelo, además, detalla los tres tipos de características de calidad en una jerarquía de factores, criterios e indicadores:

- 11 Factores (especificar): Ellos describen la visión externa del software, según las opiniones de los usuarios.
- 23 criterios de calidad (para construir): Describen la vista interna del software, como se ve por el desarrollador.
- Métricas (controlar): Se definen y se utiliza para proporcionar una escala y método para su medición.



### Modelo de Dromey

Dromey propone un modelo de calidad de productos, basado en que, reconoce que la evaluación de la calidad es diferente para cada producto. Dromey se centra en la relación entre los atributos de calidad y los sub-atributos, así como intenta conectar propiedades de productos de software (atributos externos), con la calidad de los atributos internos del software.

El modelo de Dromey tiene el propósito de trabajar con una estructura que permite construir y utilizar un modelo de calidad práctico para evaluar las etapas de Determinación de los requerimientos, Diseño e Implementación. Esta información puede ser usada para elaborar, comparar y evaluar la calidad de los productos de software. Este modelo plantea la calidad del producto por medio de la definición de sub características que pueden ser medidas y evaluadas como características.

También, permite aumentar el entendimiento respecto de la relación entre los atributos (características) y los sub atributos (sub-características) de calidad.

Dromey propone como características de calidad la Eficiencia, Confiabilidad, Facilidad de mantenimiento, Portabilidad, Facilidad de uso y Funcionalidad.

Estas características pueden ser agrupadas de acuerdo a diversos aspectos, tanto en el diseño como en la implementación: corrección, aspectos internos, aspectos del contexto y aspectos descriptivos.

Dromey propone una matriz que relaciona las características de calidad respecto de la Norma ISO 9126-1. Dicha matriz presenta un mapeo entre las características del producto y los atributos de alto nivel, el cual es utilizado en las evaluaciones del producto de software.

	Correctness Properties								Structural Properties								Modularity Properties				Descriptive Properties							
	COMPLETE	COMPLIANT	CONSISTENT	RELIABLE	INTERNALIZED	PROGRAMMABLE	VARIABLE	CONSISTENT	STRUCTURED	MODULAR	HOMOGENEOUS	RECURSIVE	RECURSIVE	RECURSIVE	RECURSIVE	ADJUSTABLE	INDIFFERENT	DIFFERENT	PARAMETERIZED	RECURSIVE	RECURSIVE	RECURSIVE	RECURSIVE	RECURSIVE	RECURSIVE	RECURSIVE	RECURSIVE	RECURSIVE
Functionality	✓	✓	✓	✓	✓	✓	✓	✓	✓																		✓	
Reliability	✓	✓	✓	✓	✓	✓	✓	✓	✓											✓	✓						✓	
Usability		✓						✓			✓															✓	✓	✓
Efficiency									✓		✓	✓	✓				✓											
Maintainability		✓			✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Portability								✓							✓											✓	✓	✓
Reusability								✓							✓	✓										✓	✓	✓

Los pasos para la aplicación del modelo de Dromey son: (1) Seleccionar el conjunto de atributos que se necesitan evaluar, (2) Realizar una lista de todos los componentes o módulos del sistema, (3) Identificar las propiedades de calidad de cada componente, (4) Determinar cómo afecta cada propiedad en los atributos de calidad y (5) Evaluar el modelo de calidad.

Las medidas de calidad del software se basan en las mediciones de **atributos** específicos, agrupados como atributos del proyecto, del proceso y del producto. Dichas medidas, cuando son analizadas, constituyen una base importante para una gestión efectiva de la calidad.

*Atributos de Procesos* son aquellos atributos relacionadas con el software. Las medidas de procesos pueden ser usadas para la mejora del desarrollo y mantenimiento del software. Tal es el caso de la efectividad de la eliminación de defectos durante el desarrollo, el patrón de aparición de defectos de prueba, y el tiempo de respuesta de los procesos mejorados. Atributos internos, entre los que podemos mencionar la duración de un proceso o de una de sus actividades, el esfuerzo asociado con el proceso o con una de sus actividades, el número de incidentes de un tipo determinado que ocurren durante el proceso o una de sus actividades, etc. En cuanto a los atributos externos, controlabilidad, observabilidad y estabilidad entre otros.

*Atributos de Productos*, componentes, entregas o documentos resultantes de una actividad de proceso. Características tales como tamaño, complejidad, características de diseño, rendimiento, y nivel de calidad se utilizan como base para las medidas de calidad aplicadas a producto. Las características medibles del producto son: Atributos externos tales como, usabilidad, eficiencia, integridad, reusabilidad, portabilidad, y testeabilidad. Los atributos internos por su lado se definen a través de tamaño del producto, longitud de las especificaciones, modularidad del diseño, acoplamiento y cohesión, complejidad del código

*Atributos de Recursos*, medidas de proyectos que describen las características y ejecución de este. Se incluyen dentro de esta categoría el número de desarrolladores de software, la estructura del personal a través del ciclo de vida del proyecto, el costo, la planificación y la productividad. Las medidas de recursos ayudan a controlar el proceso indicando cómo el proceso está usando y transformando las entradas en salidas. Los recursos incluyen cualquier entrada en la producción de software: Personal, Materiales, Herramientas, Métodos. Los atributos internos más representativos son tamaño del equipo, tiempo de experiencia, madurez de las herramientas. Los externos por su parte hacen referencia a costo, productividad, cantidad de salida / esfuerzo de entrada.

La esencia de la ingeniería de calidad de software es investigar las relaciones entre las medidas del proceso, las características del proyecto y la calidad del producto final, orientados tanto a la mejora de procesos como a la calidad del producto. Aunque para todos resulta fundamental asegurar la calidad, muchos especialistas consideran que los procesos y productos generados son los que cobran mayor importancia. Esto es debido a que generalmente no es posible modificar los recursos existentes, pero sí influir tanto en el proceso de desarrollo como en el producto final.

A continuación una selección de atributos que son comunes a los modelos presentados y propios de una evaluación de producto, con su correspondiente descripción. Es una primera aproximación a la selección de estos, luego y teniendo presente la aplicabilidad, abordaremos aquellos atributos necesarios para la evaluación de un producto de software.

**Comprensibilidad:** Capacidad de comprender el software operativo, de cara a un cambio o arreglo.

**Confiabilidad:** significa que un programa particular debe de seguir funcionando ante la presencia de errores. Los errores pueden estar relacionados al diseño, a la implementación, a la programación, o el uso de errores.

**Corrección** ¿Hace lo que se le pide? El grado en que una aplicación satisface sus especificaciones y consigue los objetivos encomendados por el cliente.

**Eficiencia:** ¿Qué recursos hardware y software necesito?. La cantidad de recursos hardware y software que necesita una aplicación para realizar las operaciones con los tiempos de respuesta adecuados.

**Escalabilidad:** La escalabilidad del software significa un aumento de la capacidad de una aplicación al agregar servidores. Además, con el escalamiento de software, la relación del costo y la capacidad agregada es casi lineal.

**Facilidad de mantenimiento** ¿Puedo localizar los fallos?. El esfuerzo requerido para localizar y reparar errores.

**Facilidad de prueba** ¿Puedo probar todas las opciones?. El esfuerzo requerido para probar una aplicación de forma que cumpla con lo especificado en los requisitos.

**Fiabilidad** ¿Lo hace de forma fiable todo el tiempo? El grado que se puede esperar de una aplicación lleve a cabo las operaciones especificadas y con la precisión requerida

**Facilidad de Aprendizaje:** Facilidad con la que nuevos usuarios desarrollan una interacción efectiva con el sistema o producto. Está relacionada con la predictibilidad, sintetización, familiaridad, la generalización de los conocimientos previos y la consistencia.

**Flexibilidad:** Relativa a la variedad de posibilidades con las que el usuario y el sistema pueden intercambiar información. También abarca la posibilidad de diálogo, la multiplicidad de vías para realizar la tarea, similitud con tareas anteriores y la optimización entre el usuario y el sistema. ¿Puedo añadir nuevas opciones?

**Integridad** ¿Puedo controlar su uso?. El grado con que puede controlarse el acceso al software o a los datos a personal no autorizado.

**Interoperabilidad** ¿Podrá comunicarse con otras aplicaciones o sistemas informáticos? El esfuerzo necesario para comunicar la aplicación con otras aplicaciones o sistemas informáticos.

**Mantenibilidad:** Se define como la capacidad de un producto software para ser modificado. Las modificaciones pueden incluir correcciones, mejoras o adaptación del

software a cambios en el entorno, en los requerimientos o en las especificaciones funcionales

**Portabilidad:** Se define como la característica que posee un software para ejecutarse en diferentes plataformas, el código fuente del software es capaz de reutilizarse en vez de crearse un nuevo código cuando el software pasa de una plataforma a otra. A mayor portabilidad menor es la dependencia del software con respecto a la plataforma.

**Reusabilidad:** Es la habilidad de los elementos de software para servir en la construcción de muchas aplicaciones diferentes. La reutilización tiene un efecto sobre todos los demás aspectos de la calidad del software, resolver el problema de reutilización significa esencialmente que la cantidad de software que será necesario desarrollar es menor, de ahí que podemos dedicar más esfuerzo (por el mismo costo total) a mejorar los otros factores, como exactitud y robustez.

**Robustez:** Es el nivel de apoyo al usuario que facilita el cumplimiento de sus objetivos. Está relacionada con la capacidad de observación del usuario, de recuperación de información y de ajuste de la tarea al usuario.

**Usabilidad:** Se refiere a la capacidad de un software de ser comprendido, aprendido, usado y ser atractivo para el usuario, en condiciones específicas de uso.

En los Modelos de Calidad, la calidad se define de forma jerárquica. Es un concepto que se deriva de un conjunto de sub-conceptos, cada uno de los cuales se va a evaluar a través de un conjunto de métricas. Tienen una estructura, por lo general, en tres niveles:

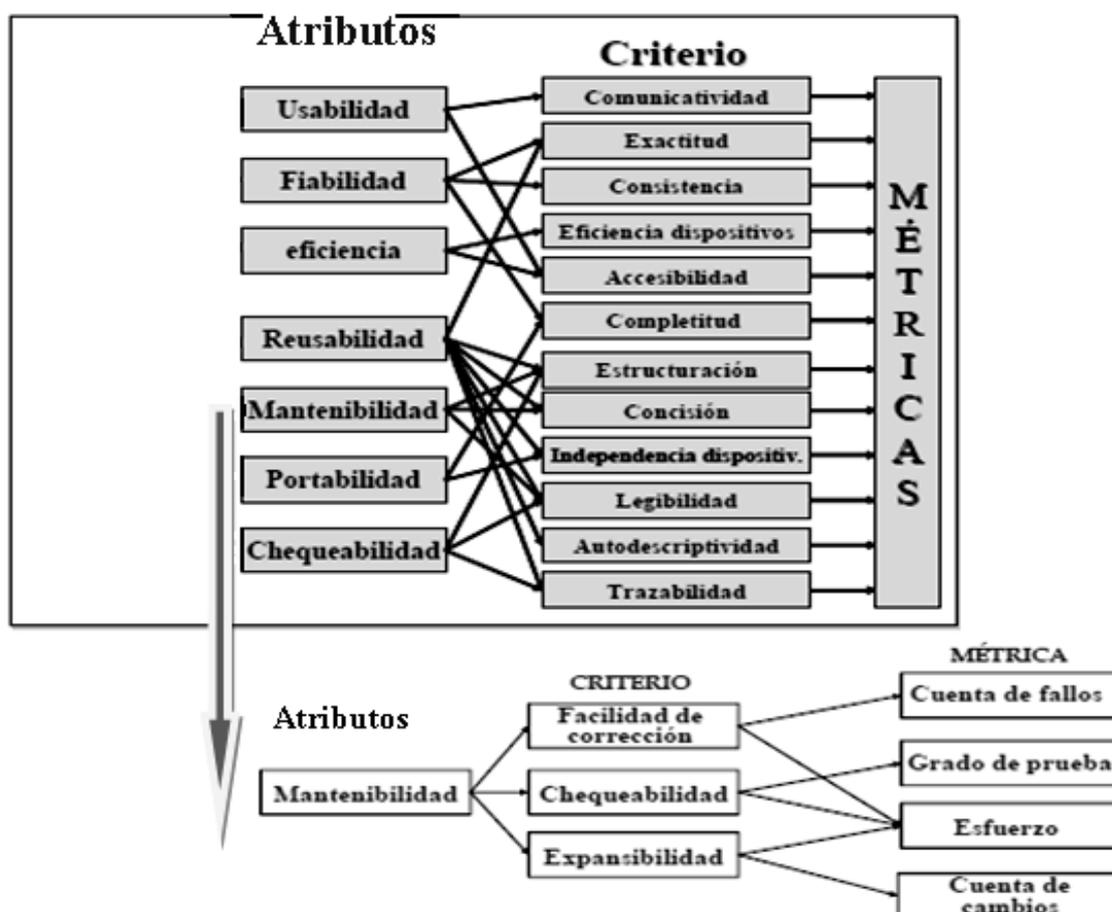
(1) En el nivel más alto de la jerarquía se encuentran los Atributos de Calidad Externos, que representan la calidad desde el punto de vista del usuario y son las características que componen la calidad.

(2) Cada uno de los Factores se descomponen en un conjunto de Atributos de Calidad Internos o Criterios de Calidad. Estos criterios son atributos que, cuando están

presentes, contribuyen al aspecto de la calidad que el factor asociado representa. Se trata de una visión de la calidad desde el punto de vista del producto de software.

(3) Para cada uno de los Criterios de Calidad se definen un conjunto de Métricas/medidas, las cuales son medidas cuantitativas de ciertas características del producto que, cuando están presentes, dan una indicación del grado en que dicho producto posee un determinado atributo de calidad.

En forma grafica se presenta a continuación una idea de la relación establecida entre atributos (también llamados atributos externos), criterios (también llamados atributos internos) y métricas o medidas.



El tamaño es uno de los atributos más medidos, no solo porque es una medida básica usada en la mayoría de las mediciones enfocadas, principalmente, al control de proyectos y la estimación de defectos, sino también porque una vez medido el tamaño, podemos medir el tiempo del desarrollo de sistemas y, una vez calculado el tiempo, podemos hacer la estimación del costo.

La complejidad, se ha usado en varios contextos, por lo tanto, este atributo fue otro de los que acumula una gran cantidad de mediciones de sus diferentes aplicaciones.

La entidad más medida es el producto de software, ya que medir el producto es mucho más sencillo que medir el proceso y el proyecto, para los que no se tienen bien definidos sus atributos. Si miramos el producto final, el hecho de tener que determinar si un software trabaja correctamente, así como analizar aspectos de mantenibilidad o robustez de acuerdo a los requerimientos del cliente, hacen esta verificación más compleja en el software, que en la industria.

El Control Estadístico de Procesos (SPC) es una aproximación estadística orientada para medir y analizar características de procesos de software y productos, de forma tal que el rendimiento de las actividades que los productos generan pueda ser gestionado, controlado y mejorado, así como que se puedan predecir.

Con el objetivo de evaluar la eficacia de la aplicación del Control Estadístico de Procesos en un contexto de evaluación del producto software, estamos transcurriendo una etapa de investigación teórica, de las características de ciertos atributos de calidad externos e internos (criterios de calidad) y sus correspondientes medidas/métricas. La intención es lograr un conocimiento más acabado de cada uno, para vincularlos especialmente a cálculos estadísticos y técnicas de gráficos de control, en la medida que sus particularidades lo permitan.

A modo de ejemplo, y en un primer acercamiento al tema, enunciamos algunos atributos con sus respectivas medidas/métricas y describimos uno de ellos en más detalle.

Atributo: Mantenibilidad, Métrica/Medida: Media de defecto: Se refiere al número de imperfecciones que puede darse por: líneas de código, definiciones de funciones o líneas en pantallas de entrada.

Atributo: Confiabilidad, Métrica/Medida: Tiempo medio de fallo: Se refiere al tiempo promedio que pasa antes de que el producto experimente un fallo.

Atributo: Mantenibilidad, Métrica/Medida: Satisfacción del cliente: generalmente consiste en una encuesta pidiendo al cliente que valore su grado de satisfacción con respecto al producto y/o sus propuestas de mejoramiento.

### Media de defecto.

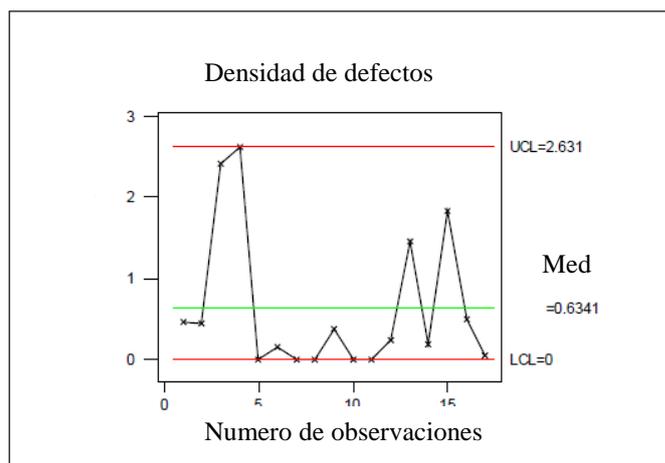
El número de defectos en un producto es una medida importante, ya que nos proporciona una referencia para medir el grado de satisfacción del cliente, el volumen de trabajo que se necesita realizar, el nivel de eficiencia de los procesos de inspección, los procesos que aún requieren inspección, y los componentes del sistema que son propensos a presentar errores. Por lo tanto, el conteo de defectos brinda evidencia no sólo de la calidad del producto, sino también de la calidad del proceso relacionado.

Como procedimiento afectado al cálculo de esta medida, se plantea trabajar con distintas muestras de funcionamiento del software, que accedan a obtener medias de defectos de cada una y en base a ellas construir una gráfica de control que permita determinar si existen desviaciones en algunos de los valores muestrales observados.

La medida resultante es la Media de defectos, que se define como:

Media de defecto = # de defectos / tamaño del producto

El análisis y la interpretación de la medida, “media de defectos” dependen de asumir que, como promedio, existe cierta expectativa a partir del conteo de defectos en relación con el tamaño del software.



Si bien se puede trabajar con las gráficas de control, para darle seguimiento a los datos relacionados con la media de defectos, se debe tener especial cuidado en las conclusiones. Las observaciones que exceden el límite superior pueden indicar un componente altamente defectuoso, o un proceso de inspección muy exitoso. De la misma forma, una medida de media de defecto muy baja puede deberse a un producto de muy alta calidad con muy pocos defectos, o a un proceso de inspección muy pobre. La determinación de las causas fundamentales requiere de un análisis detallado, el cuál debe ser llevado a cabo independientemente del análisis estadístico.

El compromiso actual está centrado en continuar con el proceso de definición de los atributos y de las medidas propias, enunciándose las características generales de cada una, las particularidades del proceso de medición, las pautas para el análisis de datos y los usos posibles de gráficas de control estadístico.

## Conclusión

Hemos expuesto a lo largo de nuestro trabajo, que muchos especialistas consideran fundamental asegurar la calidad de los procesos y productos. Que son dos aspectos que en estos días, cobran significativa importancia.

La obtención de un Software con Calidad implica la utilización de procedimientos estándares que permitan uniformar la filosofía de trabajo, tanto para la labor de desarrollo como para el Control de Calidad del producto de Software.

La calidad del producto de software abarca una calidad interna: medible a partir de las características intrínsecas, como el código fuente, una calidad externa: medible en el comportamiento del producto, como en una prueba, y la calidad en uso: durante la utilización efectiva por parte del usuario.

El objetivo no es necesariamente alcanzar una calidad perfecta, sino la necesaria y suficiente para cada contexto de uso a la hora de la entrega y del uso por parte de los usuarios.

Cada medida tiene características particulares y complejidades relacionadas con su definición, colección e interpretación. A medida que avanzamos en la investigación vamos deduciendo que es muy difícil construir una lógica general como guía estándar para la implementación de mediciones de software. Cada medida tiene su propia dinámica, características inherentes y técnicas de normalización. Los datos se coleccionan en diferentes momentos, se realizan análisis en diferentes puntos del ciclo de vida y se deben usar gráficos diferentes para la representación de cada medida. Por lo tanto, se requieren aproximaciones separadas para cada métrica a analizar.

Sabemos por ahora que en la definición de las métricas/medidas, se deben tener en cuenta tres puntos importantes para su aplicación al control estadístico de procesos. En primera instancia la importancia de entender los datos para su correcta definición; el hecho de que no es posible aplicar el SPC a todas las medidas de software, y

finalmente la estrecha correlación que debe existir entre las características de calidad definidas por los clientes para los productos y las medidas seleccionadas.

Estamos definiendo las métricas compatibles a los atributos seleccionados. Estamos también en un proceso de definición del cálculo y aplicabilidad al campo estadístico. Evaluar en cada medida la posibilidad de implementar, de forma adecuada, una gráfica de control, que actúe como auditor, demostrando no conformidades existentes en procesos/productos, es un objetivo que se está trabajando en nuestro proyecto. La utilización de las técnicas SPC no se trata de una aplicación sencilla, pero el esfuerzo puede ser justificado a partir de los beneficios esperados.

Luego de toda esta tarea, procederemos entonces a redactar un protocolo. Una manera de hacerlo será exponiendo una definición clara de los procedimientos y cálculos que orientan o guían la cuantificación de las métricas trabajadas. Somos conscientes de que el protocolo que propondremos no instaurará la forma única para todas las situaciones, constituirá un instrumento, a ser aplicado y probado, una manera de estandarización de las actividades y de descripción de las mismas, que será un tutor, una guía, al momento de medir la calidad de un producto de software. Sea cuál fuere la forma de descripción del mismo, constituirá una herramienta que el ingeniero en sistemas utilizará para facilitar y respaldar el proceso de evaluación de la calidad de los software en empresas del medio.

## Bibliografía

- Mario G. Piattini y Otros. Calidad de Sistemas Informáticos. Editorial Ra-Ma. 2006
- Roger Pressman. Ingeniería del Software: Un Enfoque Practico. McGraw-Hill. 2006
- Ian Sommerville. Ingeniería de Software. Pearson. 2005
- Alfredo Weitzenfeld. Ingeniería de Software Orientada a Objetos: Teoría y Práctica con UML y Java. Thomson Paraninfo. 2005
- Mario G. Piattini y Otros. Análisis y Diseño de Aplicaciones Informáticas de Gestión: Una perspectiva de Ingeniería. Editorial Ra-Ma. 2003
- Mario G. Piattini y Otros. Calidad en el Desarrollo y Mantenimiento del Software.. Editorial Ra-Ma. 2003
- AENOR, 1995 AENOR, UNE-EN-ISO 8402 Gestión de la calidad y aseguramiento de calidad. Vocabulario (ISO 8402:1994), AENOR, 1995
- Calvo y Fernández, 1996 Calvo-Manzano, J.A. y Fernández, L., "Hacia la calidad del software a través de la mejora de procesos", Novatica, nº 123, 1996, pp. 35-42.
- Fenton y Pfleeger, 1997 Fenton, N.E. y Pfleeger, S.L., Software Metrics, A rigorous approach; Chapman&Hall, 1997.
- Gilb, 1987 Gilb, T, Principles of Software Engineering Management; Addison Wesley, 1987.
- IEEE 1061, 1992 IEEE, IEEE Std 1061-1992. Standard for a software quality metrics methodology, IEEE, 1992.
- ISO, 1994a] ISO, ISO 9001:1994. Quality systems -- Model for quality assurance in design, development, production, installation and servicing, ISO, 1994.
- ISO, 1994b ISO, ISO 9002:1994. Quality systems -- Model for quality assurance in production, installation and servicing, ISO, 1994. [ISO 9000-3, 1997] ISO, Part 3: Guidelines for the application of ISO 9001:1994 to the development, supply, install and maintenance of computer software, ISO, 1997.
- ISO, 1998 ISO, ISO 15504-1: 1998 Software process assessment. Part-1: Concept and introductory guides, ISO, 1998.

- McCall et al., 1977 McCall J A, Richards PK y Walters GF; Factors in software quality, Vols I,II,III; US Rome Air Development Center Reports NTIS AD/A-049 014, 015, 055, 1977.
- Calero, Coral. Moraga, María Angeles. Piattini, Mario. 2010. "Calidad del Producto y Proceso de Software". Ediciones RA-MA.
- Scalone, Fernanda. 2006. Tesis de Maestría en Ingeniería en Calidad. U.T.N. FR.Bs.As. "Estudio comparativo de los modelos y estándares de calidad del software". Director: García Martínez, Ramón.
- Tello Diaz-Marioto, Inmaculada. 2009. "Formación a través de Internet. Evaluación de la Calidad". Editorial UOC.
- Escuela Superior de Informática de la Universidad de Castilla – La Mancha  
Gema de la Incera Torres
- Kan, S. H. (2002). Metrics and Models in Software Quality Engineering, Addison-Wesley.
- Gema de la Incera Torres. Medidas de calidad en proceso, producto y mantenimiento, aplicadas al Control Estadístico de Procesos. Escuela Superior de Informática de la Universidad de Castilla – La Mancha