

La calidad vista desde la ingeniería de software

Ing. Gerson Abdiel Vera Morales

gavm.sistemas@gmail.com

Ingeniero de Sistemas. Investigación de Operaciones.

Maestrante en Administración. Mención Gerencia.

Centro de Investigaciones y Desarrollo Empresarial. CIDE.

Facultad de Ciencias Económicas y Sociales.

Universidad de Los Andes. Mérida, Venezuela.

Resumen

La calidad ha estado presente desde la antigüedad, donde los egipcios dieron muestras de revisiones e inspecciones con una alta precisión, pasando por la segunda Guerra Mundial, donde comenzaron a capacitarse con mayor fuerza en el tema de la calidad, hasta nuestros días, donde las distintas técnicas y metodologías desarrolladas en el tiempo han evolucionado de manera importante, permitiendo contribuir de forma positiva en las organizaciones actuales.

La ingeniería de software como disciplina encargada de la elaboración y construcción de productos de software y prestación de los servicios asociados no escapa a las exigencias de la calidad, en tal sentido se han desarrollado estándares con la finalidad de atender lo relacionado a la forma en la que se construye y elabora el producto o se presta el servicio así como de la calidad del producto y la percepción que tienen clientes y usuarios finales del mismo.

El presente artículo hace un breve recuento histórico de la importancia de la calidad desde tiempos antiguos y cómo su influencia impactó directamente en la reciente industria del desarrollo de software, donde han surgido y evolucionado una serie de estándares y metodologías específicas del hecho del software con la finalidad de garantizar la calidad del mismo. Se realizó una compilación de distintos estándares considerados como los más influyentes de acuerdo al criterio de varios autores así como sus principales características basados tanto en la calidad del proceso de elaboración del software como en la calidad y uso del mismo.

Palabras claves: calidad, aseguramiento de la calidad, calidad de software, ingeniería de software, procesos, producto.

La Calidad vista desde la ingeniería de software.

*Ing. Gerson Abdiel Vera Morales
gavm.sistemas@gmail.com*

The quality and the software engineering

Abstract

The quality has been present since antiquity, where the ancient Egyptians showed samples of inspections and inspections with high precision, going through the Second World War, where they began to train more strongly on the issue of quality, until today, where The different techniques and methodologies developed over time have evolved significantly, allowing positive contributions in current organizations.

Software engineering as a discipline responsible for the development and construction of software products and provision of associated services does not escape the requirements of quality, in this sense standards have been developed with the purpose of addressing the related to the way in which that the product is built and elaborated or the service is provided as well as the quality of the product and the perception that customers and end users have of it.

This article gives a brief historical account of the importance of quality since ancient times and how its influence directly impacted the recent software development industry, where a series of standards and methodologies specific to the fact of software have arisen and evolved. purpose of guaranteeing the quality of it. A compilation of different standards considered as the most influential according to the criterion of several authors as well as their main characteristics based on the quality of the software development process as well as the quality and use of the same.

Keywords: quality, quality assurance, software quality, software engineering, processes, product.

La calidad vista desde la ingeniería de software

La calidad es uno de los cuatro objetivos fundamentales de las operaciones, junto con el costo, la flexibilidad y en la entrega, aún cuando la administración de la calidad es de carácter interfuncional e involucra a toda la organización, el área de operaciones tiene una responsabilidad especial en cuanto a la elaboración de un producto de calidad para el cliente. Ello requiere de la cooperación de toda la organización y una cuidadosa atención de la gerencia y control de la calidad (Schroeder *et al.*, 2011).

El aseguramiento de la calidad, por lo general se encuentra asociado a alguna forma de medición e inspección, los cuales son dos temas fundamentales de la administración de la calidad y han sido aspectos importantes de las operaciones de la producción a lo largo de la historia (Colliers y Evans, 2009). Desde las pinturas murales egipcias de alrededor de 1.450 a.C, las cuales evidencian medición e inspección, pasando por las piedras de las pirámides, que fueron cortadas de manera exacta, por mencionar tan sólo dos ejemplos de la antigüedad. El éxito de los egipcios se debió al constante uso de métodos y procedimientos bien desarrollados y dispositivos de medición precisos (Colliers y Evans, 2009).

Posteriormente, durante la revolución industrial, el uso de partes intercambiables y la división del trabajo en tareas más pequeñas, requería un control de calidad meticuloso, lo cual llevó a una dependencia de la inspección para identificar y eliminar los defectos. Con el tiempo, las organizaciones crearon departamentos de calidad separados, esta separación artificial de los trabajadores de producción de la responsabilidad por el aseguramiento de la calidad, condujo hacia una indiferencia hacia la calidad por parte de los trabajadores así como de los gerentes (Colliers y Evans, 2009).

La Segunda Guerra Mundial constituyó un nuevo hito para la evolución del aseguramiento de la calidad, muchos especialistas se capacitaron en el uso de herramientas estadísticas, volviéndose el control estadístico de la calidad muy conocido y poco a poco adoptado en todas las industrias de manufactura. Sin embargo, como la alta gerencia había delegado tanta responsabilidad por la calidad a otros, tenía poco conocimiento de la misma y cuando la crisis de la calidad llegó

La Calidad vista desde la ingeniería de software.

*Ing. Gerson Abdiel Vera Morales
gavm.sistemas@gmail.com*

unos cuantos años después, estaban mal preparados para enfrentarla (Colliers y Evans, 2009).

Durante este periodo Joseph Juran y W. Edwards Deming, dos consultores estadounidenses, presentaron técnicas de control estadístico a los japoneses para ayudarlos en sus esfuerzos de reconstrucción. Una parte significativa de su actividad educativa se concentró en la alta gerencia, en vez de especialistas de calidad independientes. Con el apoyo de los altos directivos, los japoneses integraron la calidad en todas sus organizaciones y desarrollaron una cultura de mejora continua (Colliers y Evans, 2009).

Las mejoras en la calidad japonesa eran lentas y constantes; pasaron alrededor de 20 años antes de que la calidad de los productos japoneses excediera a la de los fabricantes occidentales, para la década de los setenta, sobre todo debido a los niveles de calidad superior de sus productos, las empresas japonesas habían tenido una penetración considerable en los mercados. La reacción de la mayoría de las empresas estadounidenses fue iniciar campañas detalladas de mejora de la calidad, centradas no solo en la conformidad sino también en la mejora de la calidad del diseño (Colliers y Evans, 2009).

A medida que las organizaciones comenzaron a integrar principios de calidad a sus sistemas de administración, la noción de administración de la calidad total se volvió popular. La calidad total representaba un interés a lo largo de la cadena de valor en vez de sólo durante las operaciones de producción y la participación de cada persona y función en la organización. La calidad adquirió un nuevo significado de excelencia en el desempeño de toda la organización, en lugar de una disciplina técnica basada en la ingeniería (Colliers y Evans, 2009).

En años recientes, un nuevo interés en la calidad surgió en las salas de juntas corporativas bajo el concepto de *Six Sigma*, un enfoque centrado en el cliente y orientado a los resultados para mejorar los negocios. *Six Sigma* integra herramientas y técnicas de calidad que se han probado y validado con los años, con una orientación esencial que tiene un gran atractivo para la gerencia. (Colliers y Evans, 2009).

En la actualidad, han surgido gran cantidad de técnicas y metodologías con la finalidad de encontrar y mantener la tan apreciada calidad, es por ello que, en la

La Calidad vista desde la ingeniería de software.

*Ing. Gerson Abdiel Vera Morales
gavm.sistemas@gmail.com*

mayoría de las organizaciones, independientemente de su orientación (manufactura o servicios) se encuentran constituidos distintos sistemas de gestión que le dan vida. Dentro de estos sistemas se encuentran el sistema de gestión de la calidad, el sistema de gestión financiera, el sistema de gestión ambiental, el sistema de gestión de seguridad y otros, cada uno de estos sistemas, es un pilar dentro de la organización y el fracaso de uno de ellos podría ocasionar el fracaso total de la organización, en este sentido, el sistema de gestión de la calidad es una parte de la organización que actúa en pro de la satisfacción del cliente, por medio de la mejora de la calidad de la organización, incluyendo la mejora de sus procesos medulares, productos y servicios (Souri, 2016).

Los procesos medulares son aquellos relacionados directamente con el propósito o misión, es decir, aquellos que fueron pensados originalmente como procesos necesarios para cumplir con la razón de ser de la organización, de igual manera aplica cuando la prestación de servicios es la razón de ser del ente, en cuyo caso, el proceso medular estará formado por todas las actividades relacionadas con la prestación de ese servicio (Souri, 2016).

Al hablar de la calidad de los servicios es pertinente realizar ciertas distinciones ya que la definición y medición es del todo distinta a las de la calidad de la manufactura. La calidad del servicio entraña dimensiones que consisten en el producto expedido, el servicio tangible (explícito) y el servicio psicológico (implícito), aunque la calidad del producto que se ofrece puede medirse mediante el uso de las dimensiones de manufactura, los servicios tangible y psicológico requieren mediciones distintas. Las mediciones de la manufactura pueden ser altamente objetivas mientras que muchas medidas de servicio son perceptivas o subjetivas. La calidad de la conformidad puede calcularse a través del costo de los desperdicios y de los reprocesamientos de la fábrica (Schroeder *et al.*, 2011).

Es importante conocer cómo aplican los conceptos asociados a la calidad en la creciente industria del software y en el proceso para su construcción o elaboración. En la actualidad, el software tiene un papel dual, pues es un producto y al mismo tiempo es el canal para entregar un producto, a través de la prestación de un servicio. En su forma es un producto, el cual brinda el potencial de cómputo incorporado en el hardware de cómputo o con más amplitud en una red de

La Calidad vista desde la ingeniería de software.

*Ing. Gerson Abdiel Vera Morales
gavm.sistemas@gmail.com*

computadoras a las que se accede por medio de un hardware local, ya sea que resida en un teléfono móvil u opere en el interior de una computadora central o servidor, el software es un transformador de información: produce, administra, adquiere, modifica, despliega o transmite información que puede ser tan simple como un solo *bit* o tan compleja como una presentación con multimedios generada a partir de datos obtenidos de decenas de fuentes independientes. Por tal razón la enorme industria del software se ha convertido en un factor dominante en las economías del mundo industrializado (Pressman, 2010).

Aunque cientos de autores han desarrollado definiciones personales sobre la ingeniería del software, la propuesta por Fritz Bauer sirve como base para el análisis, indica que la ingeniería de software es el establecimiento y uso de principios fundamentales de la ingeniería con el objeto de desarrollar en forma económica, productos de software que sean confiables y que trabajen con eficiencia.

La ingeniería del software es también la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software, es decir, la aplicación de la ingeniería al software es una tecnología con varias capas, cualquier enfoque de ingeniería, incluso la de software debe basarse en un compromiso organizacional con la calidad. La administración total de la calidad, *Six Sigma* y otras filosofías similares alimentan la cultura de la mejora continua y es ésta cultura la que lleva en última instancia al desarrollo de enfoques cada vez más eficaces en la ingeniería de software. El fundamento en el que se apoya la ingeniería de software es el compromiso con la calidad. (Pressman, 2010).

Con el tiempo se han propuesto varias dimensiones y factores de la calidad del software, todos ellos tratan de definir un conjunto de características que, si se logran, permitirán elaborar un software de alta calidad, algunos estándares establecen características tales como confiabilidad, usabilidad, mantenimiento, funcionalidad y portabilidad, como indicadores de la existencia de calidad. (Pressman, 2010).

Toda organización se enfrenta al dilema de la calidad de software. En esencia, todos quieren elaborar sistemas de alta calidad, pero en un mundo dirigido por el mercado, sencillamente no se dispone del tiempo y esfuerzo requeridos para

La Calidad vista desde la ingeniería de software.

*Ing. Gerson Abdiel Vera Morales
gavm.sistemas@gmail.com*

producir software “perfecto” (Pressman, 2010). Sin importar el enfoque que se elija, la calidad tiene un costo que puede estudiarse en términos de prevención, evaluación y falla. Los costos de prevención incluyen todas las acciones de la ingeniería de software diseñadas para prevenir los defectos. Los costos de evaluación están asociados con aquellas acciones que evalúan productos de trabajo de software para determinar su calidad. Los costos de falla incluyen el precio interno de fallar y los efectos externos que precipitan la mala calidad (Pressman, 2010).

Según el estándar ISO 8402 (1986), un modelo de calidad puede definirse como el conjunto de factores de calidad y de relaciones entre ellos, que proporciona una base para la especificación de requisitos de calidad y para la evaluación de la calidad de los componentes software.

Los modelos de calidad se estructuran generalmente como una jerarquía (ya sea un árbol o un grafo dirigido), donde factores de calidad más genéricos, como eficiencia o usabilidad, se descomponen en otros más particulares, como tiempo de respuesta o facilidad de aprendizaje, probablemente en diversos niveles de descomposición. Los modelos de calidad de software pueden aplicarse en diversas actividades particulares: establecer los requisitos de calidad para la selección de un componente en base a los factores de calidad del modelo, evaluar la calidad de un componente para cada uno de los factores de calidad del modelo, comparar la calidad de distintos componentes respecto a los requisitos establecidos para un proceso de selección y redactar contratos formales, donde aparezcan explícitamente las evaluaciones de calidad de los componentes que el proveedor certifica. Normalmente, los factores de calidad que aparecen en el modelo pueden utilizarse como lista de chequeo para todas aquellas cuestiones relacionadas con la calidad de los componentes.

Desde que se formuló el concepto de modelo de calidad, se han presentado múltiples propuestas. Dichas propuestas intentan resolver entre otros, las interrogantes siguientes: ¿Cuáles son los factores de calidad que deberían formar parte de un modelo de calidad de componentes software?, ¿Cuáles son los tipos de factores de calidad en los que tiene sentido estructurar los modelos?, ¿Cómo se estructuran los modelos?, ¿Qué tipo de relaciones pueden existir entre los factores de calidad?, ¿Cómo se evalúan los factores de calidad?. A continuación se presenta

La Calidad vista desde la ingeniería de software.

*Ing. Gerson Abdiel Vera Morales
gavm.sistemas@gmail.com*

una clasificación de los tipos de modelos de calidad, las propuestas de estándares de modelos de calidad más usadas y una descripción y comparación entre las propiedades de las distintas propuestas (Carvallo *et al.*, sf).

Los modelos de calidad de software se clasifican de acuerdo con el enfoque de evaluación, ya sea a nivel de procesos, producto o calidad en uso, (Callejas *et al.*, 2017) en esta etapa conviene realizar una aclaratoria en cuanto al alcance de cada enfoque.

Desde el punto de vista de la calidad a nivel de procesos, ésta debe ser planificada y estimada desde el inicio del proyecto y posteriormente, en cada etapa del proceso de desarrollo de software, se debe llevar a cabo el control y seguimiento de los aspectos asociados a la calidad en la ejecución de los procesos para la construcción del mismo, con la finalidad de minimizar los riesgos y ofrecer soporte continuo, garantizando un óptimo nivel de cumplimiento de los factores de calidad (preestablecidos), teniendo en cuenta que si en alguna de las etapas se deja de lado la verificación de los factores y criterios es posible que, se presente deficiencia en alguno de éstos y por lo tanto disminuirá la calidad del proceso y probablemente del producto en desarrollo también (Callejas *et al.*, 2017).

Cuando la calidad es medida a nivel de producto, el objetivo principal es evaluar el cumplimiento de criterios previamente especificados para el mismo. Este enfoque está orientado a verificar el cumplimiento de las características que permitan alcanzar la satisfacción del cliente en cuanto a los requisitos definidos en las etapas iniciales del proceso de desarrollo de software (Callejas *et al.*, 2017).

Finalmente, sobre la calidad en el uso, es importante resaltar que aunque en diferentes escenarios se utilizan los términos de usabilidad y calidad en uso con el mismo propósito y de forma intercambiable, tienen significados distintos, debido principalmente a que el concepto de calidad en uso es mucho más amplio y abarca más elementos que la usabilidad y ésta última es una de las características de calidad de un producto de software. La calidad en uso se define como el conjunto de atributos relacionados con la aceptación por parte del usuario final y está basada en la eficacia, productividad, seguridad y satisfacción de acuerdo a las pautas de ISO/IEC 9126 (Callejas *et al.*, 2017).

Sobre el aseguramiento de la calidad del software, esta es una actividad de la ingeniería de software que se aplica en cada etapa del proceso de desarrollo de software de forma transversal y en paralelo al resto de las actividades. El aseguramiento de la calidad incluye procedimientos para la aplicación eficaz de métodos y herramientas, supervisa las actividades de control de calidad, tales como las revisiones técnicas y las pruebas de software, procedimientos para la administración y control del cambio y procedimientos para asegurar el cumplimiento de las normas y mecanismos de medición y elaboración de reportes. (Pressman, 2010).

Para llevar a cabo el aseguramiento de la calidad de software de manera adecuada, deben recabarse, evaluarse y divulgarse datos sobre el proceso de la ingeniería de software, los métodos estadísticos aplicados al aseguramiento de la calidad de software ayudan a mejorar la calidad del producto y del proceso de software mismo. Los modelos de confiabilidad de software amplían las mediciones, lo que permite que los datos obtenidos acerca de los defectos se extrapolen hacia las tasas de falla proyectadas y hacia la elaboración de pronósticos de confiabilidad. (Pressman, 2010).

En cuanto a los modelos a nivel de proceso, Callejas *et al.*, (2017) realizó una revisión cronológica de algunos modelos a nivel de procesos de desarrollo de software obteniendo como resultado los siguientes:

- ITIL (año 1989): desarrollado en el Reino Unido, con el fin de fortalecer la gestión gubernamental, a partir de cinco (5) elementos fundamentales: la perspectiva del negocio, entrega del servicio, soporte del servicio, manejo de la infraestructura y manejo de aplicaciones, con el propósito de ofrecer una estructura integral para prestar a la organización un servicio completo, cubriendo necesidades de apoyo de instalación, adecuación de redes, comunicaciones, hardware, servidores, sistema operativo, y software necesarios.
- ISO/IEC 15504: permite adaptar la evaluación para procesos en pequeñas y medianas empresas (pymes) y grupos de desarrollo pequeños, mediante la estructuración en seis niveles de madurez: nivel 0: organización inmadura, nivel 1: organización básica, nivel 2: organización gestionada, nivel 3:

organización establecida, nivel 4: organización predecible y nivel 5: organización optimizando. Su objetivo es llegar a que la organización logre ser madura, lo cual conlleva a que la organización tenga procesos definidos, responsabilidades definidas, predicción de resultados, productos entregados con calidad, que las entregas se den en los tiempos pactados, incrementar la productividad, clientes satisfechos y empleados felices. Esta norma, denominada “tecnologías de información: proceso de evaluación”, está constituida por cinco (5) partes. La segunda parte guía la evaluación del proceso y la aplicación del proceso de evaluación para el mejoramiento y determinación de la capacidad, precisa los requisitos mínimos para realizar una evaluación que asegure un nivel de consistencia y capacidad de repetición y que los resultados de la evaluación sean objetivos, imparciales, repetibles, consistentes y representativos. Identifica el marco de trabajo de medida para la capacidad del proceso y los requisitos para el modelo de procesos de referencia, el modelo de evaluación de procesos y la verificación de la conformidad del proceso de evaluación. El modelo del proceso de evaluación contiene una dimensión del proceso y una dimensión de la capacidad del proceso (Pino *et al.*, 2005).

- Dromey (año 1995): la finalidad de este modelo es evaluar varias etapas del proceso de desarrollo como: levantamiento de requisitos, diseño e implementación. Se estructura con características y subcaracterísticas de calidad; propone tres modelos distintos para cada etapa de construcción del producto: modelo de requerimientos, modelo de diseño y modelo de calidad de la implementación, a partir de la evaluación establecida en cinco etapas, para características como: eficiencia, confiabilidad, mantenibilidad, portabilidad, facilidad de uso y funcionalidad (Scalone, 2006).
- *Personal Software Process* (PSP por sus siglas en inglés): este modelo propuesto en el año 1995 está enfocado al desarrollo profesional del ingeniero, fomentando una adecuada administración de calidad de los proyectos de desarrollo, reducción de defectos del producto, estimación y planeación del trabajo (Vargas, 2010).

- *Team Software Process* (TSP) (año 1996): TSP por sus siglas en inglés, es la fase posterior de PSP, está diseñado para el trabajo de equipos de desarrollo de software autodirigidos, que se orienta al desarrollo de producto con el mínimo de defectos en tiempo y costos estimados. Cuenta con planes detallados y procesos como revisiones personales, inspecciones e índices de desempeño de calidad y el fomento de la integración del equipo (Mondragón, 2011). El PSP y el TSP incluyen reconocidas técnicas para el asentamiento de una base de información útil en la obtención de indicadores, pero aún así requieren de mucho trabajo por parte del ingeniero para lograr llevar a cabo el control necesario de sí mismo y del equipo respectivamente (Lugo *et al*, 2009).
- IEEE / EIA 12207 (año 1996): establece un marco común para los procesos del ciclo de vida del software, con una terminología bien definida, que puede ser referenciada por la industria del software. Está conformada por procesos, actividades y tareas que se aplican en la adquisición de sistemas y productos de software y servicios, para el suministro, desarrollo, operación, mantenimiento y eliminación de los productos de software y la parte de software de un sistema, ya sea realizado internamente o externamente a una organización. La IEEE 12207 proporciona procesos para definir, mejorar y controlar los procesos del ciclo de vida del software. El marco descrito por el estándar está diseñado para ser adaptado a toda organización y proyecto (IEEE SA-12207, 2008). El estándar aplica los principios relacionados con la gestión total, este estándar trata todas las actividades incluidas las relacionadas con la calidad, como parte integral del ciclo de vida del software, Estas actividades afines con la calidad son asignadas a cada proceso. Cada proceso está equipado con un plan “do check-act” (PDCA), por lo tanto a cada proceso y al personal encargado de llevarlo a cabo se le asignan sus actividades y procesos relacionados con la calidad incluyendo las evaluaciones (Gaibor, 2015).
- Cobit 4.0 (año 1996): se caracteriza por ser orientado a negocios y procesos, además de ser basado en controles, trabaja con siete criterios de información que son definidos como requerimientos de control del negocio: efectividad,

eficiencia, confidencialidad, integridad, disponibilidad, cumplimiento y confiabilidad (IT Governance Institute, 2005).

- CMMI (*Capability Maturity Model Integration* por sus siglas en inglés, año 2000): es de los modelos más utilizados en las empresas de construcción de software, con el propósito de verificar el cumplimiento de estándares de calidad a partir de la medición con niveles de madurez. Este modelo se representa de dos maneras: escalonada y continua, donde el modelo escalonado está dirigido al software y permite clasificar las organizaciones en cinco tipos de nivel establecidos: inicial, gestionado, definido, gestionado cuantitativamente y en optimización. Por su parte el modelo continuo se enfoca al análisis de la capacidad de cada proceso inmerso en las áreas de la ingeniería de sistemas y lo clasifica en uno de los siguientes seis niveles: nivel 0: incompleto: nivel 1: ejecutado, nivel 2: gestionado, nivel 3: definido, nivel 4: cuantitativamente gestionado y nivel 5: en optimización (Petrie *et al* 2009).
- ISO/IEC 20000-1:2005 (año 2005): el objetivo principal de esta norma es el de avalar que la prestación de servicios gestionados de tecnologías de información de una empresa cuentan con la calidad necesaria para brindar dichos servicios a los clientes. Se subdivide en dos partes: “Especificaciones”, publicada como ISO 20000- 1:2005 y “Código de buenas prácticas” publicada como ISO 20000-2:2005, promueve la adopción de un enfoque de procesos integrados, para una provisión eficaz de servicios gestionados que satisfaga los requisitos del negocio y de los clientes. Con la intención de desarrollar una guía para la aplicación de la norma ISO 9001 a la gestión de servicios de tecnologías de información, ISO inició en 2008 un nuevo proyecto denominado ISO/IEC NP 90006 (Mesquida *et al*, 2010).

De igual manera Callejas *et al* (2017) y Fuertes (2002) realizaron en distintos trabajos revisiones cronológicas de algunos modelos a nivel de productos empleados para la evaluación del software desde esta perspectiva.

- McCall (año 1977): uno de los más renombrados predecesores de los modelos de calidad actuales es el modelo McCall, que fue presentado por Jim McCall en 1977. Este modelo se basa en tres perspectivas generales que

La Calidad vista desde la ingeniería de software.

*Ing. Gerson Abdiel Vera Morales
gavm.sistemas@gmail.com*

son utilizadas para definir e identificar la calidad de un producto de software: revisión del producto, transición del producto y operación del producto. Los once criterios base, son: exactitud, confiabilidad, eficiencia, integridad, usabilidad, mantenibilidad, *testeabilidad*, flexibilidad, portabilidad, reusabilidad e interoperabilidad (Córdoba *et al*, sf).

- Modelo de Arthur (año 1985): L. J. Arthur presentó una pequeña variación al modelo de McCall consistente en añadir tres nuevos criterios, así como en modificar las relaciones existentes entre éstos y factores. Los nuevos criterios son: auditabilidad, complejidad y seguridad (Fuertes, 2002).
- GQM o *Goal Question Metric*, por sus siglas en inglés (año 1984): modelo cuyo objetivo es el de evaluar distintos atributos de calidad del producto, partiendo de los objetivos y las preguntas respecto a qué hace una organización para mejorar y en base a esto se establecen las métricas correspondientes (Lugo *et al*, 2009).
- Boehm (año 1986): el modelo McCall sirvió de base para el modelo de calidad Boehm, otro predecesor de los modelos de calidad actuales, creado por Barry W. Boehm. Este modelo fue el primero en proponer la evaluación de la calidad del software por medio de métodos automáticos y cuantitativos, es un modelo incremental, dividido en regiones de tareas y estas a su vez en conjuntos de tareas, las cuales se ajustan a la cantidad de iteraciones que el equipo defina, y cada iteración se divide en cuatro sectores: planeación, análisis de riesgo, ingeniería y evaluación. (Córdoba *et al*, sf).
- FURPS (año 1987): posteriormente Robert Grady, en 1987, creó el modelo FURPS, no tan conocido como los mencionados anteriormente pero basado en los mismos principios. Su principal aportación es que divide los factores principales de evaluación en dos grupos: los basados en requerimientos funcionales y los basados en requerimientos no funcionales (Córdoba *et al*, sf).
- GILB (año 1988): T. Gilb definió una jerarquía de los atributos de un sistema que para él resultaban habitualmente interesantes en el campo de la ingeniería del software. Gilb llamaba atributo a una medida de la calidad de un sistema. Dicha medida podía variar entre ciertos límites, siendo aún el

sistema aceptable para el usuario. El objetivo fundamental del ingeniero de software sería identificar los atributos críticos y los límites de cada atributo entre los cuales el sistema sigue siendo aceptable. Estos atributos se aplican a la lógica, a los datos, a la documentación, a la interfaz y demás aspectos del software. Los atributos del modelo son: capacidad de trabajo, adaptabilidad, disponibilidad y utilizabilidad, cada uno de los cuales se divide, a su vez, en sub atributos (Fuentes, 2002).

- Modelo de Deutsch y Willis (año 1988): publicaron una jerarquía similar a la de McCall, aunque incluía nuevos factores y criterios, así como nuevas relaciones entre éstos. La jerarquía parte de que las necesidades del usuario pueden descomponerse en necesidades operacionales y de mantenimiento. Las necesidades operacionales tienen que ver con la capacidad del software para llevar a cabo las tareas que se supone que debe realizar. Las necesidades de mantenimiento tienen relación con la modificación del software, con el fin de ayudar al usuario. Dentro de las necesidades operacionales, la funcionalidad trata de lo que hace el software al ejecutarse, mientras que la realización trata de lo bien que lo hace. En cuanto a las necesidades de mantenimiento, el cambio trata de las modificaciones del software para corregir errores, adaptarse a nuevos entornos o añadir nuevas funcionalidades, mientras que la gestión tiene que ver con la planificación para cambiar, controlar versiones, pruebas e instalación (Fuentes, 2002).
- Modelo de Schulmeyer (año 1991): Schulmeyer publicó una serie de indicadores de la calidad del software. Estos indicadores de la calidad no deben ser interpretados como medidas, sino que sirven de base para dichas medidas. Los indicadores serían de ayuda pues podrían dar una idea de la tendencia de la calidad (adecuación a requisitos) en el desarrollo del software. Los indicadores son: completitud de las pruebas, completitud, densidad de defectos, densidad de fallos, documentación, estructura del diseño: se utiliza para determinar la simplicidad o claridad del diseño, suficiencia de las pruebas. Schulmeyer se sale de la línea habitual y descompone la calidad en un único nivel de unos pocos indicadores, que

pueden emplearse para evaluar la calidad del desarrollo, basándose en los paradigmas tradicionales (Fuertes, 2002).

- IEEE 1061 (año 1998): tiene como objetivo la definición de métricas de software y su uso en la evaluación de componentes software. Fue aprobado en 1992 y revisado y modificado en 1998. Propone la construcción de modelos de calidad a medida adaptados a cada proyecto. No fija ningún factor de calidad, pero sí una clasificación de los factores de los que debe constar un modelo en un nivel más alto y abstracto de factores, que deben descomponerse en subfactores, que a su vez se descomponen en métricas.
- ISO/IEC 9126: tiene como objetivo la definición de un modelo de calidad y su uso como marco para la evaluación de software. Los modelos de calidad concordantes con este estándar pertenecen a la categoría de modelos mixtos, ya que el estándar propone una jerarquía de factores de calidad clasificados como características, subcaracterísticas y atributos según su grado de abstracción, entre los que se propone un conjunto de factores de partida compuestos de seis (6) características y veintisiete (27) subcaracterísticas. El estándar distingue entre calidad interna y calidad externa, e introduce también el concepto de calidad en uso. La calidad interna tiene como objetivo medir la calidad del software mediante factores medibles durante su desarrollo. La calidad externa pretende medir la calidad del software teniendo en cuenta el comportamiento de este software en un sistema del cual forme parte. Finalmente, la calidad en uso corresponde a la calidad del software desde el punto de vista de un usuario.

El ISO/IEC 9126 original fue sustituido en 2001 por dos estándares relacionados, el ISO/IEC 9126 de calidad del software y el ISO/IEC 14598 de evaluación de productos software. La versión de 2001 del ISO/IEC 9126 consta de cuatro partes: 9126-1 (2001), presenta un modelo de calidad, que es común para medir la calidad interna y externa, y uno distinto para medir la calidad en uso; 9126-2 (2003), presenta posibles métricas externas para atributos de calidad externos; 9126-3 (2003), presenta posibles métricas para atributos de calidad internos; y 9126-4 (2004), presenta posibles métricas para evaluar atributos de calidad en uso. Cabe destacar que en este cambio,

las sub características mencionadas anteriormente pasaron de ser recomendadas en un anexo, a formar parte del estándar (Carvallo *et al*, sf).

Sólo la primera parte de la norma ISO 9126-1 (2001) es un estándar aprobado y publicado, siendo los restantes informes que componen la parte identificada como Reportes Técnicos (*Technical Report TR*). A continuación se definen las características descritas en la ISO/IEC 9126 (2001):

- Usabilidad: capacidad del producto software de ser entendido, aprendido y usado por los usuarios bajo condiciones específicas.
 - Funcionalidad: capacidad del producto software de proporcionar funciones que ejecuten las necesidades explícitas e implícitas de los usuarios cuando el software es usado bajo condiciones específicas.
 - Confiabilidad: capacidad del producto software de mantener un nivel especificado de rendimiento cuando es usado bajo condiciones específicas.
 - Eficiencia: representa la relación entre el grado de rendimiento del sitio y la cantidad de recursos (tiempo, espacio, entre otros) usados bajo ciertas condiciones.
 - Mantenimiento: capacidad del producto software de ser modificado y probado.
 - Portabilidad: capacidad del producto software de ser transferido de un ambiente a otro. (Alfonzo y Mariño, 2013).
- Modelo de Gillies (año 1992): A. C. Gillies presentó su modelo de calidad que se diferenciaba de otros trabajos previos en el sentido de que consideraba todos los criterios como un subconjunto de corrección, dividiéndola en corrección funcional y empresarial. Esto requería la incorporación de nuevos criterios asociados con la corrección desde el punto de vista de los negocios. Según el autor, este modo de ver el modelo le permitiría reflejar tanto el punto de vista de calidad del diseñador como el del usuario (Fuertes, 2002).
 - Modelo REBOOT: surge del proyecto ESPRIT, está basado en la división habitual de la calidad en factores, criterios y métricas, así como en las relaciones entre sí representadas mediante una jerarquía. (Fuertes 2002).

- WebQEM (año 1998): es una metodología de evaluación de calidad de sitios web (*Web-site Quality Evaluation Method* por sus siglas en inglés), diseñada para la evaluación siguiendo seis fases: planificación y programación de la evaluación de calidad, definición y especificación de requerimientos de calidad, definición e implementación de la evaluación elemental, definición e implementación de la evaluación global, análisis de resultados, conclusión y documentación, validación de métricas. Entre las actividades que plantea la metodología WebQEM, se encuentra el diseño de indicadores parciales/globales para obtener el grado de cumplimiento global de las características de alto nivel que se estén evaluando. Para realizar esto, WebQEM utiliza el método *Logic Scoring of Preference* (LSP por sus siglas en inglés), el cual es un método cuantitativo basado en técnicas de puntuación y lógica continua de preferencias. Básicamente, LSP permite establecer criterios de evaluación, especificando las propiedades esperadas de un sistema. En este punto, todos los valores de los indicadores elementales pueden agruparse adecuadamente diseñando una estructura de agregación por niveles, que permite obtener una preferencia (valor de indicador global/parcial) de acuerdo a las necesidades del usuario y el punto de vista de la evaluación. (Gallardo *et al.*, sf).
- Estándar ISO/IEC 25000:2005 (año 2005): su propósito es guiar el desarrollo con los requisitos y la evaluación de atributos de calidad, teniendo como referencia la adecuación funcional, la eficiencia de desempeño, la compatibilidad, la capacidad de uso, la fiabilidad, la seguridad, la mantenibilidad y la portabilidad. Los aspectos más importantes en el desarrollo de software bajo este enfoque son la calidad del producto y del proceso. ISO/IEC 25000, proporciona una guía para el uso de las nuevas series de estándares internacionales, llamados Requisitos y Evaluación de Calidad de Productos de Software (SQuaRE). Constituyen una serie de normas basadas en la ISO 9126 y en la ISO 14598 y su objetivo principal es guiar el desarrollo de los productos de software con la especificación y evaluación de requisitos de calidad (Portal ISO 25000).

La familia ISO 25000 está orientada al producto software, permitiendo definir el modelo de calidad y el proceso a seguir para evaluar dicho producto. La familia de normas SQuaRE está compuesta por cinco (5) divisiones: ISO 2500n: Gestión de la calidad, ISO 2501n: Modelo de calidad, ISO 2502n: Medida de la calidad, ISO 2503n: Requisitos de calidad e ISO 2504n: Evaluación de la calidad. El estándar ISO/IEC 25000 (2005), contiene una explicación sobre el proceso de transición entre el estándar ISO/IEC 9126, las series 14598 y SQuaRE. También presenta información sobre cómo utilizar la norma ISO/IEC 9126 y la serie 14598 en su forma anterior. Ofrece términos y definiciones, modelos referencia, guía general, guías de división individual y los estándares para fines de especificación, planificación y gestión, medición y evaluación. (Alfonzo y Mariño, 2013).

- El estándar ISO/IEC 25010:2011: reemplaza y actualiza el estándar ISO 9126-1 (2001). Define un modelo de calidad en uso que se compone de cinco (5) características (algunas de las cuales se subdividen en subcaracterísticas). Se relacionan con el resultado de la interacción cuando un producto se emplea en un contexto particular de uso. Un modelo de calidad del producto que se compone de ocho (8) características (que se subdividen en subcaracterísticas). Se refieren a propiedades estáticas de software y las propiedades dinámicas del sistema informático. El modelo es aplicable a los productos de software y sistemas informáticos. Las características definidas por ambos modelos son relevantes para todos los productos de software y sistemas informáticos. Las características y subcaracterísticas proporcionan coherencia terminológica para especificar, medir y evaluar la calidad del producto software y sistemas informáticos. El modelo de calidad de producto abarca cualidades internas y externas del sistema y está compuesto por ocho (8) características y treinta y un (31) subcaracterísticas.

Algunos modelos de calidad clásicos han sido la base para los más recientes y han permitido que los modelos actuales se consolidan como los más completos con base en la evolución del software, para así optimizar los procesos de las

organizaciones y garantizar que se cumple con criterios o estándares que respaldan la calidad de la gestión de procesos del negocio. (Callejas *et al.*, 2017).

Es importante que las empresas cuyo objeto o razón de ser sea el desarrollo de software, se certifiquen bajo alguna norma o estándar, el que mejor se adapte a sus características y objetivos organizacionales, ya que esto permite que la misma tenga una mejor posición, reconocimiento y demanda en el mercado, al estar avalada por alguna entidad competente se garantiza un nivel de satisfacción mayor para los clientes y mejora su competitividad a nivel internacional.

Referencias bibliográficas

Alfonzo, P., Mariño, S., (2013). *Los estándares internacionales y su importancia para la industria del software*. Consultado el: 26/06/2018. Disponible en: <http://www.cyta.com.ar/ta1202/v12n2a3.htm>.

Callejas, M., Alarcón, A., Álvarez, A. *Modelos de calidad del software, un estado del arte*. En: *Entramado*. Enero - Junio, 2017. vol. 13, no. 1, p. 236-250, Consultado el: 26/06/2018. Disponible en: <http://dx.doi.org/10.18041/entramado.2017v13n1.25125>.

Carvalho, J., Franch, X., Quer, C. (sf). *Calidad de componentes de software*. Consultado el: 28/06/2018. Disponible en: <http://www.essi.upc.edu/~franch/papers/libro-calidad-cap-10-jpc-xf-cq-10-version-preliminar.pdf>.

Collier, D., Evans, J., (2009). *Administración de operaciones. Bienes, Servicios y Cadenas de Valor*. Segunda edición. CENGAGE Learning. México.

Córdoba, J., Cachero, C., Calero, C., Marhuenda, Y., (sf). *Modelo de calidad para portales bancarios*. Consultado el: 28/06/2018. Disponible en: https://www.researchgate.net/profile/Julio_Cordoba_Retana/publication/228869245_Modelo_de_Calidad_para_Portales_Bancarios/links/545aefd00cf25c508c31a2e1.pdf.

Fuertes, J., (2002). *Modelo de calidad para el software orientado por objetos*. Tesis doctoral. Universidad Politécnica de Madrid. Consultado el: 28/06/2018. Disponible en: http://oa.upm.es/34988/1/TD_Fuertes_JOSE_LUIS.pdf.

Gaibor, J.,(2015). *Determinación del cumplimiento de las metodologías SCRUM y XP con relación al estándar IEEE-12207 aplicado al sistema de control de*

proveeduría en la cacech. Tesis de grado. Consultado el: 28/06/2018.
Disponible en:<http://dspace.esPOCH.edu.ec/handle/123456789/4339>.

Gallardo, C., Funes, A., Ahumada, H., (sf). *Modelo integral para la evaluación de la calidad de la accesibilidad al contenido web*. Consultado el: 28/06/2018.
Disponible en:
http://sedici.unlp.edu.ar/bitstream/handle/10915/54094/Documento_completo.pdf-PDFA.pdf?sequence=1.

IT Governance Institute (2006). *Cobit 4.0*. Consultado el: 28/06/2018. Disponible en:
https://www.gob.mx/cms/uploads/attachment/file/82967/CobiT4_Espanol.pdf

Lugo, J., García A., Delgado R., (2009). *Gestión de indicadores en proyectos de software. Perspectivas actuales y futuras*. Revista Cubana de Ciencias Informáticas 3 (Julio-Diciembre). Consultado el: 28 de junio de 2018.
Disponible en: <http://www.redalyc.org/articulo.oa?id=378343637002>.

Mesquida, A., Mas, A., Esperança, C. (2010). *Sistema de Gestión Integrado según las Normas ISO 9001, ISO/IEC 20000 e ISO/IEC 27001*. REICIS. Revista Española de Innovación, Calidad e Ingeniería del Software. 6 (Noviembre-Sin mes). Consultado el: 28/06/2018. Disponible en:
<http://www.redalyc.org/articulo.oa?id=92218768002>.

Mondragón, O., (2011). *Integrando TSP y CMMI: Lo mejor de dos mundos*. Consultado el: 28/06/2018. Disponible en:
<https://sg.com.mx/revista/31/integrando-tsp-cmmi>.

Pino, F. J., García, F., Ruiz, F., Piattini, M. (2005). *Adaptación de las Normas ISO/IEC 12207: 2002 e ISO/IEC 15504: 2003 para la evaluación de la madurez de procesos software en países en desarrollo*. In JISBD (pp. 187-194). Consultado el: 28/06/2018. Disponible en:
https://www.researchgate.net/profile/Felix_Garcia2/publication/221595530

La Calidad vista desde la ingeniería de software.

Ing. Gerson Abdiel Vera Morales
gavm.sistemas@gmail.com

[Adaptacion de las normas ISOIEC 12207 2002 e ISOIEC 15504 2003 para la evaluacion de la madurez de procesos software en paises en desarrollo/links/0912f50afe7ac2e1ee000000.pdf.](#)

Pressman, R., (2010). *Ingeniería de software. Un enfoque práctico*. Séptima edición. Mc Graw Hill.

Petrie, M., Medina, V., Méndez., G., (2009). *Modelo de Registro y Acreditación de Instituciones de Educación Superior basado en el Modelo CMMI*. In Seventh LACCEI Latin American and Caribbean Conference for Engineering and Technology. Consultado el: 28/06/2018. Disponible en:http://laccei.org/LACCEI2009-Venezuela/Papers/Acc116_LarrondoPetrie.pdf

Scalone, F. (2006). *Estudio comparativo de los modelos y estándares de calidad del software*. Tesis Ingeniería de Calidad. Buenos Aires: Universidad Tecnológica Nacional Regional de Buenos Aires.. 488 p. Consultado el: 28/06/2018. disponible en:<http://laboratorios.fi.uba.ar/lsi/scalone-tesis-maestria-ingenieria-en-calidad.PDF>.

Schroeder, R., Meyer, S., Rungtusanatham, J. (2011). *Administración de operaciones. Conceptos y casos contemporáneos*. Quinta edición. Mc Graw Hill.

Souri, A., (2016). *Implantación y gestión de la Norma ISO 9001:2015. Una guía paso a paso para implantar y mantener cada requisito de la Norma ISO 9001:2015*. Editorial Melvin C.A. Venezuela.

Vargas, F., Soto, D. "Introduciendo PSP (Personal Software Process) en el aula". En: Revista Colombiana de Tecnologías de Avanzada. 2010. vol. 2 , no. 16. Consultado el: 28/06/2018. Disponible

La Calidad vista desde la ingeniería de software.

Ing. Gerson Abdiel Vera Morales
gavm.sistemas@gmail.com

en:http://www.unipamplona.edu.co/unipamplona/portallG/home_10/recursos/general/pag_contenido/publicaciones/revista_tec_avanzada/2010/numero2/09112010/rcta_16.pdf.

Velazco, A. (2016). *Modelo en espiral. Introducción Boehm*. Consultado el 28/07/2018. Disponible en: https://www.academia.edu/7972897/Modelo_espiral_INTRODUCCION_Boehm