

Estándares que contribuyen al desarrollo y entrega de productos de Software de Calidad

Ana, Chacón-Luna¹; Jorge, Rodas-Silva²; Mariuxi, Vinueza-Morales³

Resumen

Las TIC aportan a los países para la realización de progresos importantes como el cumplimiento de los objetivos del Milenio. En este sentido el presente artículo tiene como objetivo resaltar la importancia de la calidad en el desarrollo de software. Al hablar de calidad del software involucra la necesidad de describir parámetros que permitan establecer niveles mínimos, que un producto debe cumplir, para que se considere de excelente calidad. La problemática que se presenta es que la mayoría de las características que debe cumplir, para ser considerado de excelente calidad, no son fácilmente cuantificables. En virtud de ello se han desarrollado varios trabajos que establecen propuestas para el desarrollo de softwares con calidad, cumpliendo con las características básicas de todo sistema FURPS (Funcionalidad, Usabilidad, Reliabilidad; Performance, Supportability), además se han desarrollado estándares de evaluación de la calidad del producto y de la calidad de proceso de desarrollo de software. En el presente trabajo se expondrá sobre las metodologías Capability Maturity Model Integration (CMMI) e ISO/IEC 9000, puesto que a nivel mundial son las que mayormente se aplican. Además se abordará la situación de las empresas ecuatorianas en cuanto al desarrollo de softwares de calidad.

Palabras Clave: software, desarrollo de software, evaluación de la calidad.

Standards that contribute to the development and delivery of high quality software products

Abstract

ICTs contribute to countries in the implementation of major developments as the fulfillment of the Millennium Development Goals. In this sense, this article aims to highlight the importance of quality in software development. To measure quality of software it is necessary to describe the parameters that establish minimum standards that a product must meet to be considered quality. The problem that arises is that most of the features that the software must meet to be considered quality are not easily quantifiable. Therefore this work provides various proposals for the development of quality software, meeting the basic features of any system FURPS (Functionality, Usability, Reliability, Performance, Supportability). Additional standards are proposed to assess product quality and quality of software development process. In this paper we will present the methodologies Capability Maturity Model Integration (CMMI) and ISO/IEC 9000, which are those that mostly apply worldwide. Also the situation of Ecuadorian companies in the development of quality software is addressed.

Keywords: software, software development, quality assessment

Recibido: 15 de agosto de 2014

Aceptado: 31 de marzo de 2015

¹Máster en Gerencia de Tecnologías de la Información. Docente Titular de la Universidad Estatal de Milagro (UNEMI), Ecuador, Facultad Ciencias de la Ingeniería. evitachacon@gmail.com; achacon1@unemi.edu.ec

²Máster en Gerencia de Tecnologías de la Información. Docente Titular de la Universidad Estatal de Milagro (UNEMI), Ecuador, Facultad Ciencias de la Ingeniería. jorge.rodass@gmail.com; jrodass@unemi.edu.ec

³Máster en Administración y Dirección de Empresas. Docente en la carrera de Ingeniería en Sistemas Computacionales, Facultad Ciencias de la Ingeniería, Universidad Estatal de Milagro, UNEMI. yovinueza@gmail.com; mvinuezam@unemi.edu.ec

I. INTRODUCCIÓN

La predisposición de las organizaciones es brindar excelente calidad en las operaciones y servicios ofrecidos, garantizando de esta manera la eficiencia de su quehacer diario; y es así que el desarrollo de softwares no se encuentra aislado de estas exigencias, de tal manera los directores de proyectos emplean actualmente metodologías, a través de las cuales, se busca reducir costos, evitar retrasos en la planificación, mejorar el tiempo de dedicación y la calidad en el desarrollo de softwares. *“La producción de un sistema software con calidad involucra la utilización de metodologías, procedimientos y estándares para la realización del análisis, diseño, desarrollo y prueba del producto que permitan delinear la forma de trabajo, en merced de lograr mayor confiabilidad, mantenibilidad y facilidad de prueba, a la vez que elevan la productividad, tanto para la labor de desarrollo como para el control de la calidad del software”* [1].

El aseguramiento de la calidad del software (ACS) consiste en controlar que los productos y procesos del desarrollo de software cumplan con estándares de completitud y calidad. Existen dos formas de obtenerla: la primera es prevenir la falta de calidad, para ello se deberá definir normas, estándares, métodos y **técnicas apropiadas durante los procesos del ciclo de vida**. La segunda es detectar y corregir la falta de calidad, es decir; corregir los errores en el código, en el diseño, en manuales de usuarios, o código complejo mal documentado, a través de la **evaluación de procesos**, mejoramiento de

procesos, revisiones y, testeo de software [2].

Hay que considerar que las actividades de fortalecimiento de la calidad deben ser planificadas, con sus respectivas asignaciones de recursos humanos, materiales y tecnológicos etc. Es decir, para asegurar la calidad hay que invertir dinero, sin embargo, la ausencia de calidad también tiene un precio. Joseph Juran, planteó en 1951 *el análisis de costos relacionados con la calidad en su libro Quality Control Handbook*, él diferencia 3 tipos de costos de aseguramiento de la calidad (Figura 1) [3]:

La pérdida de ingresos, las oportunidades de mercado perdidas, los pedidos tramitados incorrectamente y los errores de facturación, son factores todos ellos que hacen desaprovechar dinero a la empresa. Y hay que tener en cuenta también el coste de la insatisfacción de los clientes, intangible pero real [4].

II. DESARROLLO

Para determinar el posicionamiento, la evolución y la importancia que tiene el desarrollo de softwares en Ecuador, se realizó una revisión del estado del arte sobre el crecimiento en dicho sector y las metodologías utilizadas en mayor cantidad. Obteniendo como antecedente un estudio realizado, en el año 2011, por la Asociación Ecuatoriana de Software (AESOFT), sobre análisis a nivel mundial del mercado del software, donde se evidencia que América ocupa el primer puesto en cuanto a softwares y servicios, seguido por el continente europeo, Asia y el resto del mundo; en la Tabla 1 se muestra dichas cifras.

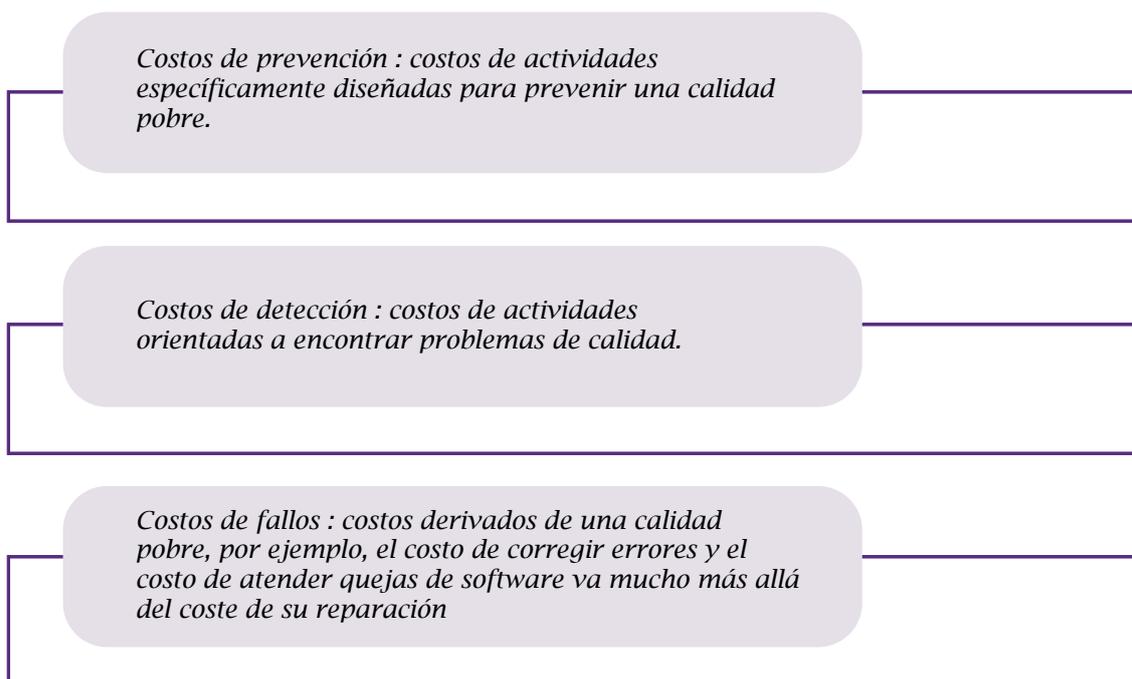


Figura 1. Costos de aseguramiento de la calidad

Tabla 1. Principales Mercados a nivel Global

	%SHARE	
	Software y servicios	Computadoras y periféricos
Américas	40.3%	35.6%
Europa	27.1%	36.7%
Asia-Pacífico	29.5%	22.4%
Resto del Mundo	3.1%	5.3%

Fuente. Datamonitor

Tomando como punto de partida las estadísticas anteriormente mencionadas, se establece que el sector del software en el Ecuador es calificado como un sector en crecimiento económico. En el año 2010 la AESOFT publicó cifras que dieron una idea de la situación real del país, donde el total de ventas por software realizado en Ecuador sobrepasaba los 200 millones de dólares en el año 2008, en el Figura 2 se evidencia este crecimiento [5].

Estas cifras se fortalecen cuando en el año 2009 el informe emitido por el Servicio de Rentas Internas (SRI) indicaba que existieron ingresos totales por el sector de Software y servicios de \$ 260 millones, valores que se han incrementado proporcionalmente, como muestra lo reportado por la consultora TATA, que en el año 2010 realizó el estudio de los ingresos para el mismo sector, donde se estimó que llegó a \$ 276,20 millones de ingresos totales [4]. Como se ha indicado es evidente que el sector de software ha presentado un progreso en los últimos años,

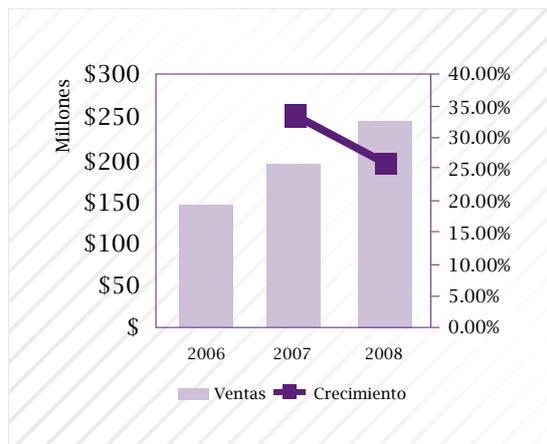


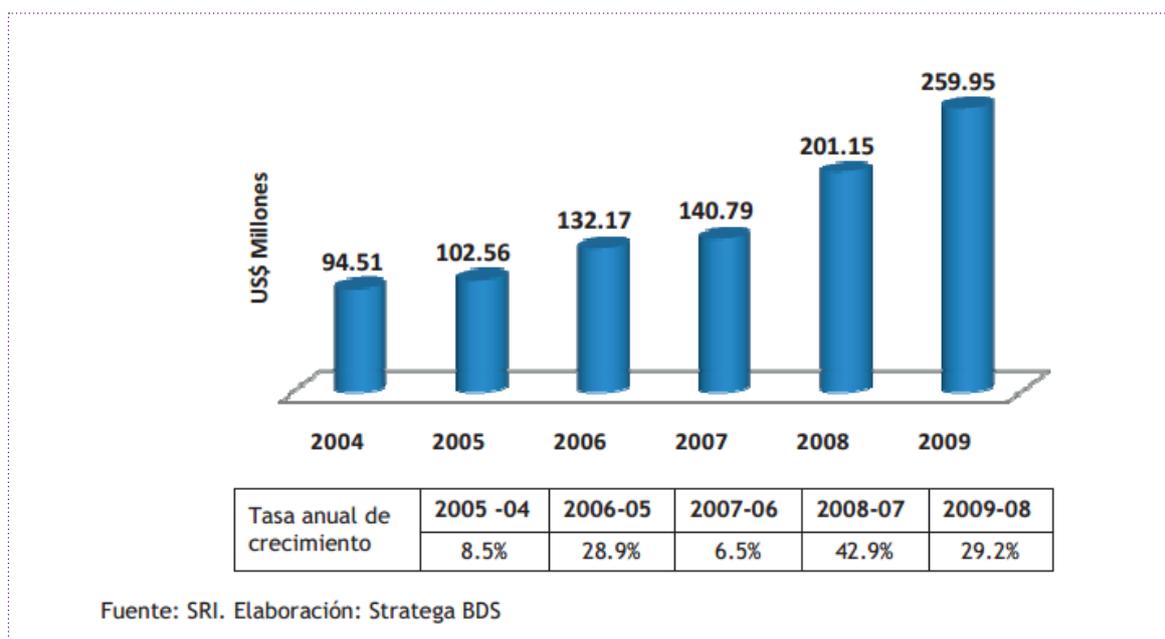
Figura 2. Ventas de la Industria del Software

Fuente. Asociación Ecuatoriana del Software, 2010

basándose en un estudio realizado por AESOFT puede asegurarse que ha alcanzado una tasa de crecimiento del 22,4% desde el año 2004 al 2009 [4]. Ver Figura 3.

Es innegable que el desarrollo del software en Ecuador se ha incrementado considerablemente y es así que en la actualidad de acuerdo a los datos de la Superintendencia de Compañías y Valores del Ecuador, existen 636 empresas registradas a nivel nacional que se dedican a Actividades de Programación Informática. Las mismas que se encuentran enmarcadas en el código J6201. En la Tabla 2 se muestra el número de compañías relacionadas con esta actividad y las ciudades donde estas se encuentran asentadas [6].

Como se observa en la Tabla 2 las ciudades donde, en mayor grado, se desarrolla software son Quito y Guayaquil,



Fuente: SRI. Elaboración: Stratega BDS

Figura 3. Ingresos Totales en el Sector de Software

Fuente. SRI. Elaborado por Stratega BDS.

notamos que en el cantón Milagro existen 4 empresas dedicadas al desarrollo de software, lo que evidencia que el cantón también aporta al crecimiento de este sector, por ello se cree necesario dar un aporte mediante la recopilación de las metodologías de procesos aplicadas al desarrollo de software de calidad más usadas en empresas locales.

Tabla 2. Empresas que se dedican a Programación Informática

CIUDADES	NUMERO DE COMPAÑÍAS	PORCENTAJE
QUITO	364	57,23%
GUAYAQUIL	190	29,87%
CUENCA	24	3,77%
LOJA	9	1,42%
AMBATO	7	1,10%
MANTA	6	0,94%
RIOBAMBA	5	0,79%
SAMBORONDÓN	5	0,79%
IBARRA	4	0,63%
MILAGRO	4	0,63%
OTROS	18	2,83%
TOTAL	636	100%

Fuente. Superintendencia de Compañías y Valores

Muy a menudo se discursa acerca del desarrollo de software de calidad, pero ¿Qué es calidad de software? Según Pressman (2002) la calidad del software es “la concordancia con los requisitos funcionales y de rendimiento explícitamente establecidos, con los estándares de desarrollo explícitamente documentados y con las características implícitas que se espera de todo software desarrollado profesionalmente” [7]. Según la Norma ISO 8402 se considera como: “La totalidad de las características de un producto o servicio que le confieren su aptitud para satisfacer las necesidades declaradas o implícitas” [8].

Sabemos que el software se encuentra inmerso en muchas áreas agilizando los procesos desarrollados por las organizaciones, su utilización contribuye al aumento de velocidad de procesamiento, incrementa la productividad, etc. Se puede deducir que la implementación de software es de vital importancia en una organización, en relación a ello Jenkins & Pedro (2000) manifiestan “La ventaja competitiva de cualquier empresa en tecnología de información radica en el software y el *peopleware* que posea, y no en el hardware que adquiera” [9]. Esta cita exhorta a tomar en consideración la importancia de la implementación de un software de calidad que contribuya a la ejecución de los objetivos organizacionales; entendiéndose por *peopleware* el desarrollo de software

por personas y equipos productivos mediante aprendizaje organizado y cultura organizacional.

Para custodiar la buena o excelente calidad del software hay que considerar ciertas características, tales como:

- El software es un producto intangible y su calidad también lo es.
- El software no perderá calidad con el pasar del tiempo, es decir, todos los problemas que aparezcan durante el mantenimiento, en realidad estaban desde el principio, no se crean nuevos errores.
- Cada error en el software implica una falla en el diseño o en el proceso o en el desarrollo.

De manera general podemos indicar que el software siempre se enfrentará a problemas como:

- El software es dinámico e iterativo a lo largo de su ciclo de vida, puesto que evolucionan de una versión a otra para mejorar sus características.
- Es difícil conseguir productos totalmente depurados.
- Normalmente no se terminan en los plazos, ni con los costos pronosticados, y en muchas ocasiones tampoco cumplen los niveles deseables de los requisitos especificados por el usuario [10].

Sin embargo ¿Es realmente posible desarrollar softwares con excelente calidad? La respuesta a esta interrogante es: Sí. Para el desarrollo de software con calidad se deberá emplear modelos de calidad y entre las distintas actividades para la implantación del control de calidad en el desarrollo de software es necesario:

1. Aplicación de metodología y técnicas de desarrollo.
2. Reutilización de procesos de revisión formales.
3. Prueba del software.
4. Ajustes a los estándares de desarrollo.
5. Control de cambios, mediciones y recopilación de información.
6. Gestión de informes sobre el control de calidad [7].

El software en la actualidad se ha incorporado en áreas financieras, comerciales, de servicio, de salud, etcétera, prácticamente en todas las actividades cotidianas. Como se ha indicado la utilización de un software supone mejoras y agilidad en el trabajo desarrollado, sin embargo ¿quién no se ha enfrentado a situaciones como: no poder llamar desde un teléfono móvil, recibir facturas con errores, tener problemas para hacer una compra o reservar tickets aéreos?, o cuando alguna función del automóvil no se lleva a cabo correctamente, muchas veces estas anomalías se deben precisamente a un error de software, y son estos sucesos los que tergiversan el objetivo del software generando desventajas cuantiosas, en aspectos

económicos, sociales, medioambientales e incluso ocasionando pérdidas humanas.

La falta de buena calidad del software afecta la credibilidad de las organizaciones. Durante los últimos años se han producido errores de software que por su trascendencia o espectacularidad forman ya parte de la historia del sector tecnológico. Estos son algunos de ellos [11]:

La Conquista del Espacio: Fobos 1, Mariner 1 y Ariane 5

Son fracasos del Instituto de Investigaciones Espaciales de la Unión Soviética, la NASA y la Agencia Espacial Europea, respectivamente. Por pequeños errores de programación que causaron grandes pérdidas al momento de su implementación, y es así que 25 años después la Fobos 1 aún siga en órbita sin enviar señal a la tierra. La NASA destruyó a Mariner 1 pues se iba a estrellar en el Océano Pacífico, Ariane 5 explotó tan solo a los 40 segundos del lanzamiento

La máquina de Radioterapia Therac-25

Desarrollada por la empresa estatal canadiense de energía nuclear, causó la muerte de tres pacientes entre 1985 y 1987 por suministrar sobredosis de radiación. Se concluyó que las malas prácticas en desarrollo, mal análisis de requerimientos y mal diseño del software causó ese grave error.

Error en los procesadores Pentium

En 1994 un profesor de la Universidad de Lynchburg descubrió un error en la unidad de coma flotante del procesador Intel Pentium. El error generaba valores erróneos en algunas operaciones de división. Intel tuvo que cambiar todos los procesadores que tenían este defecto lo que supone (ocasionó) pérdidas cuantiosas.

De los casos explicados puede deducirse que la falta de buena calidad del software es un factor determinante del fracaso, por haber omitido metodologías que velen por su calidad.

¿De dónde proceden los problemas de software?

De acuerdo a un estudio realizado por “Software Quality”, Mordechai Ben- Benachem y Cary S. Marliss aseguran que esto se debe a:

- Requerimientos insuficientes comprendidos 50%
- Diseño no comprendido o incorrectamente trasladado de los requerimientos: 30%
- Codificación (error de programación o diseño mal comprendido): 20%

Por otro lado, más del 50% del personal técnico de software y un 70% de los directores de software tienen una

formación deficiente en control de calidad [12].

Para confrontar estas cifras existe un estudio realizado a 26 empresas del sector de desarrollo de software, 20 en la ciudad de Quito y 6 en Guayaquil, quienes proporcionaron información respecto a la capacitación que se les provee al personal de desarrollo. Los resultados evidencian que el 58% de las empresas realizan capacitaciones en forma trimestral y semestral, pues consideran que este es un factor crítico en el desempeño laboral, por otra parte el 21% de las empresas realizan capacitaciones mensuales, proporcionando beneficios como aumento de la productividad y calidad del trabajo que realizan, además aumento en la rentabilidad de la empresa y mejora del conocimiento de las actividades a realizar y por parte del personal de desarrollo. Únicamente dos empresas no realizan capacitaciones a sus empleados, lo que corresponde a un 8% [13], ver Figura 4.

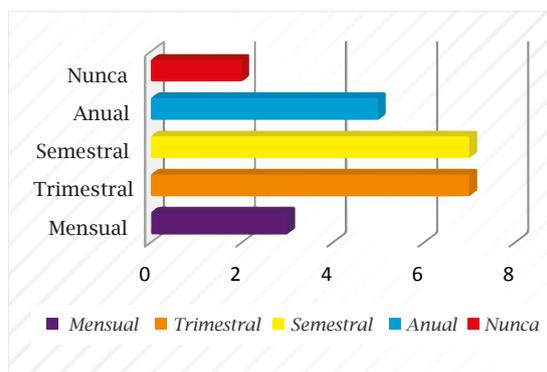


Figura 4. Capacitación del personal de desarrollo en empresas Ecuatorianas

Fuente. Análisis de la aplicación de los modelos de calidad de software en las empresas desarrolladoras asentadas en Quito y Guayaquil

El análisis realizado en este estudio da a conocer que los directivos de empresas ecuatorianas, dedicadas al desarrollo de software, están conscientes de la necesidad de aplicación de metodologías de calidad en los proyectos ejecutados por sus desarrolladores, de tal manera que cerca del 79% brindan capacitación constante a sus colaboradores.

Para disminuir los inconvenientes generados en el desarrollo de software, además de exigir rigurosidad en la etapa de pruebas y mantenimiento donde supone dejar libre de errores al software, antes de la puesta en producción o entrega del producto al cliente, es imprescindible que se evalúe también el proceso de desarrollo, apoyándose en modelos de buenas prácticas reconocidas. Hay que tener claro que los estándares o metodologías, delimitan criterios de desarrollo que orientan la forma en que se aplica la Ingeniería del Software, no obstante si no se aplica ninguna metodología para el desarrollo de software siempre habrá falta de buena calidad en el mismo.

Normalmente las áreas claves sobre las que trabajan estas buenas prácticas son:

- Requerimientos de software
- Diseño de software
- Construcción del software
- Testeo del software
- Mantenimiento del software
- Gestión de la configuración del software
- Gestión de la ingeniería del software
- Proceso de ingeniería del software
- Métodos y herramientas de ingeniería del software
- Calidad del software [14].

Hay que decidir cuál será el dominio de aplicación del modelo o estándar de calidad a implementar, ya que se lo podrá hacer a nivel de proceso y a nivel de producto.

Calidad de proceso

El objetivo final del modelo de procesos es lograr una representación clara de los procesos reales de desarrollo, con la cual se podrá trabajar para planificar las mejoras a incluir en cada uno de esos procesos. La mejora del producto final pasa, según estos modelos, por la mejora de los procesos que llevan a su creación. La adopción del modelo o metodología adecuados podrá realizar esta mejora con una correcta implantación, dotando implícitamente al producto final de una calidad. Entre los modelos o estándares a nivel de proceso se pueden mencionar ISO 9001:2000 (ISO 90003:2004), ISO/IEC 15504, CMMI, TickIT, ISO 20000, Bootstrap, SwTQM (basado en EFQM y CMMI), etcétera.

Calidad de producto

La estrategia en este caso se basa en el establecimiento de varios puntos de testeo y verificación sobre todas las etapas del ciclo de vida (requisitos, análisis, diseño y construcción), además de la última etapa de pruebas finales. Las técnicas a emplear suelen ser las siguientes:

- Detección temprana de errores mediante la anticipación del testing.
- Utilización de técnicas especializadas, como los test estáticos, dinámicos, pruebas unitarias, pruebas de caja negra y caja blanca, etc.
- Incorporación a los equipos de trabajo de grupos especializados en testing.

Entre los modelos o estándares a nivel de producto se pueden mencionar TPI/TMAP, el Modelo de Boehm, el Modelo de Gilb, el Modelo de Dromey, la norma ISO 9126-1, el Modelo de McCall, WebQEM, ISO 25000, Portal Quality Model (PQM) y otros. Mientras que los modelos a nivel de proceso son independientes de la tecnología, los modelos y estándares a nivel de producto surgen o se

actualizan de acuerdo a la evolución tecnológica [14].

El 8% de las empresas ecuatorianas encuestadas indicó aplicar CMMI y un 4% la Norma ISO/IEC 12007, mientras el 40% aseguró conocer de los estándares y normas de calidad, sin embargo utilizan sus propios métodos, desarrollados en base a la experiencia obtenida en el proceso de creación de software, el 48% afirmó no aplicar ninguna norma de calidad. Estas cifras son preocupantes, pues cerca de la mitad de las empresas encuestadas en dicho estudio manifestó, en el año 2012, no haber adoptado modelos de calidad para el desarrollo de sus productos, demostrando deficiencia en comparación con otros países en cuanto a desarrollo de software se refiere [13].

Se cree necesario ahondar en estas temáticas para conocer las bondades que ofrecen dichas metodologías a la hora de desarrollo de software.

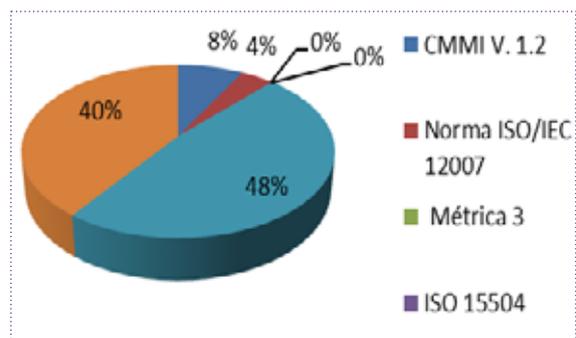


Figura 5. Aplicación de estándares de calidad en empresas de desarrollo Ecuatorianas.

Fuente. Análisis de la aplicación de los modelos de calidad de software en las empresas desarrolladoras asentadas en Quito y Guayaquil

CMMI

El modelo CMMI for Development (CMMI-DEV) Capability Maturity Model Integration reúne un conjunto de buenas prácticas de ingeniería de software que facilitan el mejoramiento gradual de los procesos de gestión de proyectos, gestión de procesos, ingeniería y soporte.

El modelo CMMI ayuda a resolver problemas típicos de las organizaciones, tales como:

- Productos y servicios que no satisfacen los requerimientos del cliente
- Demoras y baja rentabilidad en los proyectos
- Altos costos operacionales
- Baja productividad
- Bajos niveles de innovación y desmotivación del personal [15].

Es decir CMMI es un indicador de la madurez de la empresa, por medio del cual se mejora los procesos. El objetivo es optimizar la usabilidad de los modelos de madurez, complementando diferentes modelos en un solo marco de trabajo (framework).

De acuerdo a Software Engineering Institute CMMI

“es una estrategia de mejora de procesos que proporciona a las organizaciones los elementos esenciales que en última instancia, mejoran su rendimiento. CMMI puede ser usada para guiar la mejora de procesos en un proyecto, una división o una organización entera. Ayuda a integrar funciones tradicionalmente separadas de la organización, establece objetivos y prioridades de mejora de procesos, proporciona una guía para los procesos de calidad y proporciona un punto de referencia para evaluar los procesos actuales” [16].

Condiciones para el éxito y principios de mejoramiento

Se presenta un listado ordenado de las condiciones comunes para el éxito de iniciativas de mejoramiento de procesos, y es representativo para diferentes contextos:

1. Compromiso de la alta gerencia.
2. Liderazgo operacional.
3. Comunicación.
4. Entrenamiento de los líderes.
5. Entrenamiento de los empleados.
6. Medición del progreso.
7. Manejo de la iniciativa como un proyecto.
8. Alineamiento de la iniciativa con la estrategia global.
9. Uso de herramientas de soporte.
10. Articulación de beneficios.
11. Mejoramiento incremental.
12. Involucramiento de los empleados.
13. Cambio del sistema de administración de la productividad [17].

Descripción del modelo CMMI (CMMI v1.1, CMMI for Development)

CMMI Establece 5 niveles de Madurez para clasificar a las organizaciones, en función de qué áreas de procesos consiguen sus objetivos y se gestionan con principios de ingeniería. Es lo que se denomina un modelo escalonado, o centrado en la madurez de la organización. Los niveles son los siguientes:

1. **Inicial.** Las organizaciones en este nivel no disponen de un ambiente sólido para el desarrollo y mantenimiento de software. Aunque se utilicen técnicas correctas de ingeniería, los esfuerzos se ven minimizados por la falta de planificación. El éxito de los proyectos se basa la mayoría de las veces en el esfuerzo personal, aunque a menudo se producen fracasos y casi siempre retrasos y sobrecostos, haciendo que el resultado de los proyectos sea impredecible.
2. **Repetible/Gestionado/Administrado.** En este nivel las organizaciones disponen de prácticas institucionales de gestión de proyectos, existiendo métricas básicas y un razonable seguimiento de la calidad. La relación con subcontratistas y clientes está

gestionada sistemáticamente.

3. **Definido.** Además de una buena gestión de proyectos, a este nivel las organizaciones disponen de correctos procedimientos de coordinación entre grupos, formación del personal, técnicas de ingeniería más detallada y un nivel más avanzado de métricas en los procesos. Implementándose bajo estas pautas técnicas de revisión por pares (peer reviews).
4. **Gestionado cuantitativamente.** Se caracteriza porque las organizaciones disponen de un conjunto de métricas significativas de calidad y productividad, que se usan de modo sistemático para la toma de decisiones y la gestión de riesgos. El software resultante al utilizar estas medidas es de alta calidad.
5. **Optimizado.** La organización completa está volcada en la mejora continua de los procesos. Se hace un uso intensivo de las métricas y se gestiona el proceso de innovación [14].

De acuerdo al informe emitido en Marzo de 2014 por CMMI Institute, se evidencia el crecimiento entre los años 2007 a 2013 sobre los cinco niveles de madurez del modelo CMMI, reflejados en el número de organizaciones que han acogido esta metodología. Ver Figura 5.

CMMI se está convirtiendo poco a poco en un estándar que puede ser usado para promocionar la capacidad de desarrollar software de alta criticidad, o que puede dar una ventaja competitiva si se desea participar en proyectos de alta complejidad y riesgo, que por obvias razones, tienen un alto precio y muy buenas ganancias. Por ejemplo, Boeing, General Dynamics, IBM, Lockheed Martin, Motorola, Raytheon o Toshiba son algunas de las empresas que han alcanzado el nivel 5 de CMMI, lo que les abre las puertas a proyectos de decenas o cientos de millones de dólares [18].

¿Quiénes están certificados en CMMI en el mundo?

De acuerdo al informe emitido en Marzo de 2014 por CMMI Institute, el año 2013, el crecimiento de empresas certificadas con CMMI a nivel mundial lo encabezó China, con cerca del 48% de las certificaciones obtenidas, mientras que los países que más sobresalen en Latinoamérica son Brasil 218 y México 208. Ver Figura 7.

ISO 9000:2000

La familia de normas ISO 9000 es un conjunto de normas de calidad establecidas por la Organización Internacional para la Estandarización (ISO), aplicables en cualquier tipo de organización.

Algunas de las bondades relevantes de esta Norma son:

- La aproximación de la gestión basada en el proceso.
- Compatibilidad con otros sistemas de gestión (p.ej. ISO 14000).

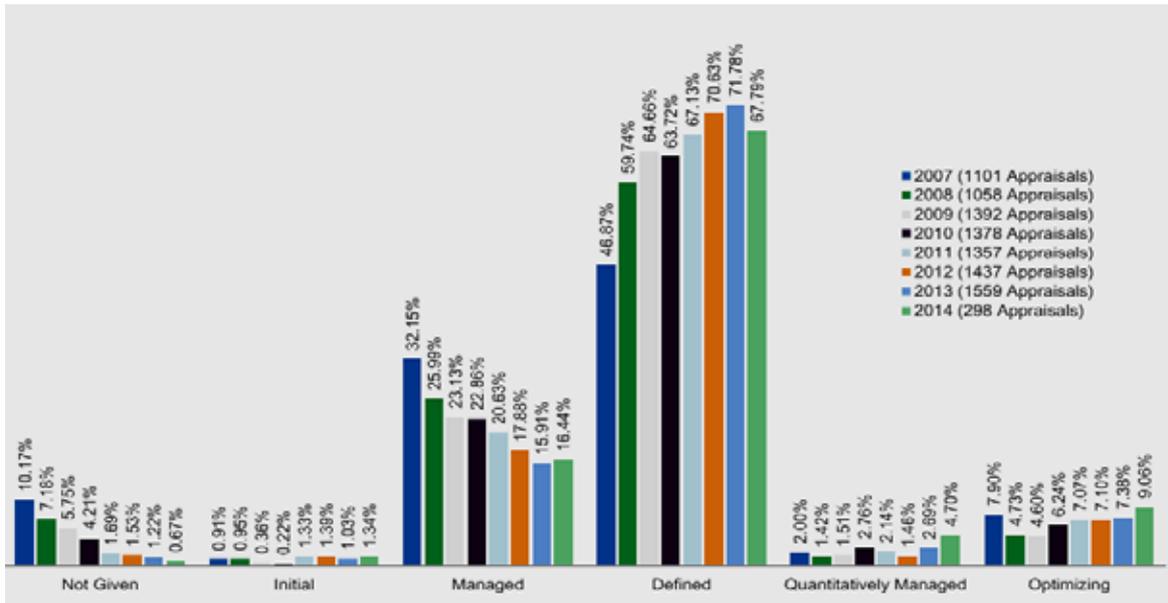


Figura 6. Tendencias en las fases de CMMI

Fuente: CMMI

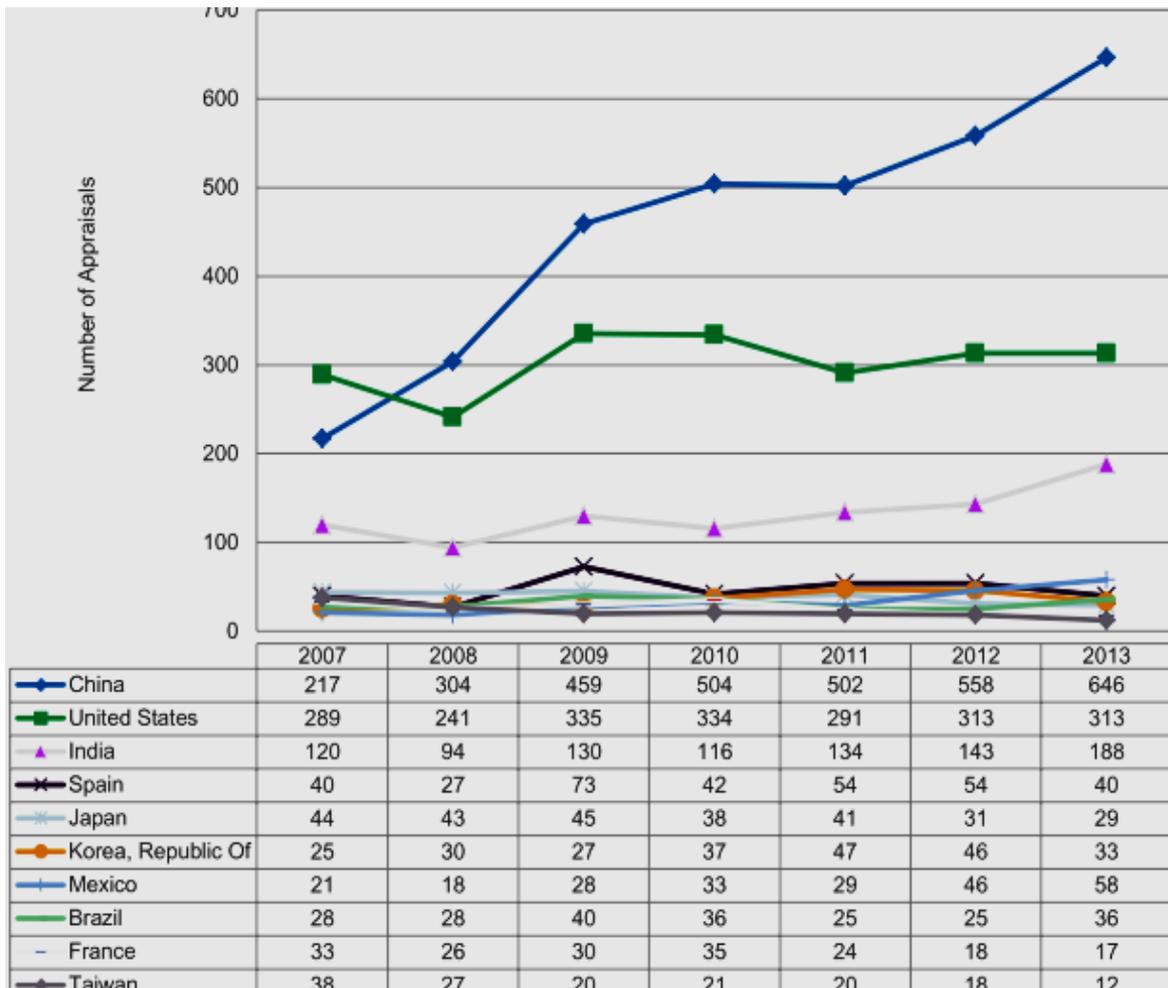


Figura 7. Tendencias de adopción de CMMI

Fuente: CMMI

- Inclusión de requisitos para la mejora continua del sistema de calidad.
- Coincidencia con las necesidades de los participantes externos (p.ej. clientes, proveedores, etc.).
- Facilidad de uso para el usuario y el cliente (Inteco, 2008) [14].
- Además comprende las normas ISO 9001, 9044, 19011.

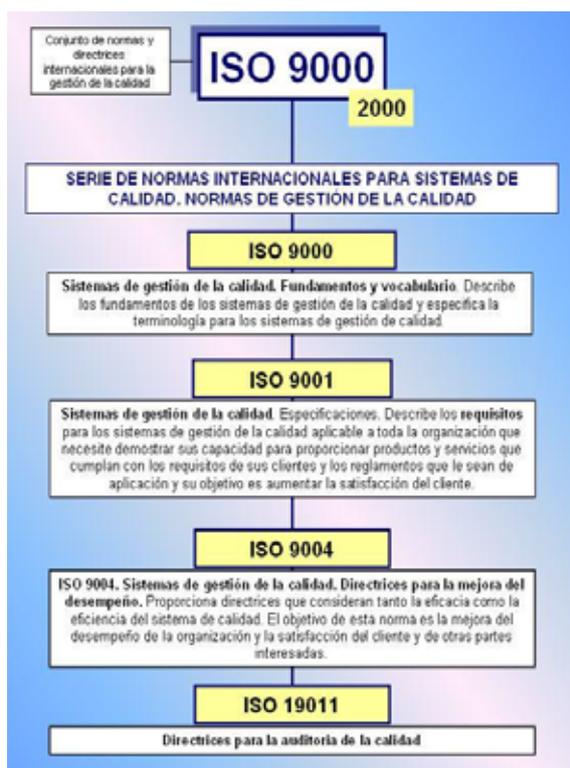


Figura 8. Norma ISO 9000:2000
 Fuente: <http://endrino.pntic.mec.es/jhem0027/calidad/normalizacion/normalizacion.html>

El estándar se basa en un conjunto de Principios de Gestión de la Calidad, tales como: el enfoque permanente al cliente final, el liderazgo, la implicación de todo el personal, el enfoque a procesos, la mejora continua, el enfoque objetivo hacia la toma de decisiones, las relaciones mutuamente beneficiosas con los proveedores.

Esta norma es aplicable a:

1. Organizaciones que buscan ventajas por medio de la implementación de un sistema de gestión de la calidad.
2. Organizaciones que buscan la confianza de sus proveedores en que sus requisitos para los productos serán satisfechos.
3. Usuarios de los productos.
4. Aquellos interesados en el entendimiento mutuo de la terminología utilizada en la gestión de la calidad: (por ejemplo: proveedores, clientes, entes reguladores),
5. Todos aquellos que perteneciendo o no a la

organización evalúan o auditan el sistema de gestión de la calidad para determinar su conformidad con los requisitos de la norma ISO 9001:2000, (por ejemplo: auditores, entes reguladores, organismos de certificación / registro),

6. Todos aquellos que perteneciendo a la organización o no asesoran o dan formación sobre el sistema de gestión de la calidad adecuado para dicha organización,
7. Aquellos quienes desarrollan normas relacionadas [19].

La ISO 9001:2000 es la norma que provee los requisitos básicos que debe cumplir una organización para implementar un sistema de gestión de calidad, y es la única de las anteriormente mencionadas que es certificable en la actualidad.

Estructura de ISO 9001:2000

La estructura de ISO 9001:2000 tiene muchas semejanzas al círculo de Deming, mediante el ciclo PDCA (Plan-Do-Check-Act). La norma se estructura en los siguientes capítulos: Ver Figura 9.

III. CONCLUSIONES

De acuerdo a la revisión bibliográfica se puede evidenciar que las empresas de desarrollo de software en Ecuador, en su mayoría, no aplican normas y estándares de calidad de software [13]. Estos resultados no son muy alentadores ya que la industria del software no ha sido potencializada, para obtener mayores beneficios, conociendo que el mercado de software está creciendo de manera acelerada y cada vez más se lo integra a distintas actividades de la economía de los países y vida de las personas.

A continuación se indican algunos aspectos relevantes del estudio realizado.

Actualmente los requerimientos de nuevos productos tienen que ver con la calidad, sin embargo la evolución, de la tecnología es un desafío que deben enfrentar las empresas de desarrollo de software.

Por la competitividad en este sector es necesario que las empresas obtengan certificaciones de calidad del software, para que de esta manera poder competir internacionalmente.

De acuerdo a las exigencias del mercado, una empresa puede requerir certificar más de un estándar, esto se determina mediante una evaluación a la empresa considerando entre otros aspectos: los objetivos empresariales, requerimientos del mercado al cual va dirigido sus productos, características propias del software desarrollado, entre otros.

IV. REFERENCIAS

[1] Hernandez, S., & Comp-, T. R. (2008). La importancia

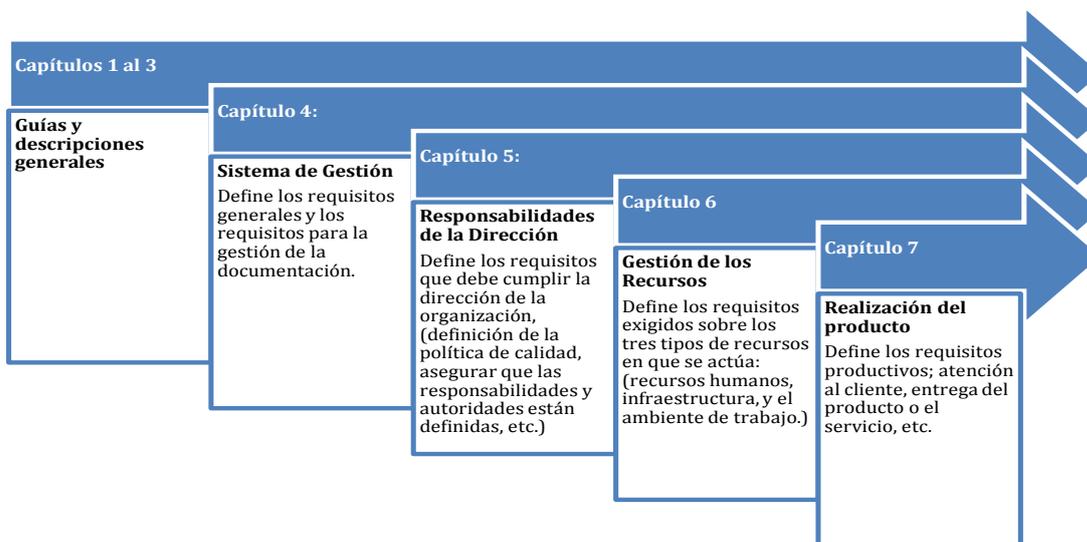


Figura 9. Costos de Aseguramiento de la Calidad

Fuente. <http://endrino.pntic.mec.es/jhem0027/calidad/normalizacion/normalizacion.html>

- de la calidad en el desarrollo de productos de software.
- [2] Mollineda, R. y Vos, T. (2004). Calidad y Testeo del Software, 15. Extraído de : <http://www.google.com.ec/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CBsQFjAA&url=http%3A%2F%2Fwww.iti.es%2Fmedia%2Fabout%2Fdocs%2Ftic%2F01%2F2003-07-calidad.pdf&ei=VhE4VMaRHtLesATuiILwAg&usg=AFQjCNFPiQ1xfyYy6QQeiLvb34w1WAVE9QAESOFT.> (2011). Estudio de Mercado del Sector de Software y Hardware en Ecuador.
 - [3] Mollineda, R. y Vos: *Ibíd*, p 15.
 - [4] AESOFT. (2011). Estudio de Mercado del Sector de Software y Hardware en Ecuador.
 - [5] INEC. (2012). Resumen del Entorno TIC en el Ecuador a 2011.
 - [6] Valores, S. I. de C. y. (2014). No Title. Retrieved from <http://181.198.3.71/portal/cgi-bin/cognos.cgi>
 - [7] Pressman, R. (2002). Ingeniería del Software. Un enfoque práctico. (M. G. Hill, Ed.) (Mc Graw Hill).
 - [8] Hernandez, S., & Comp-, T. R. (2008). La importancia de la calidad en el desarrollo de productos de software.
 - [9] Jenkins, M., & Pedro, S. (2000). para para el y Mantenimiento Mantenimiento Calidad.
 - [10] César, S. V. (2014). Tesis Análisis de la norma ISOIEC 27001.
 - [11] López Torets, J. M. (2013). Los Cinco Grandes Errores de la Historia del Software. Retrieved from <http://www.mtp.es/noticias/264-los-cinco-grandes-errores-de-la-historia-del-software>
 - [12] Hernandez, S., & Comp-, T. R. (2008). La importancia de la calidad en el desarrollo de productos de software.
 - [13] [16]. Vinueza, M. (2012). Análisis de la Aplicación de los Modelos de Calidad del Software.
 - [14] Inteco. (2008). Estudio sobre la certificación de la calidad como medio para impulsar la industria de desarrollo del software en España.
 - [15] Arboleda, H., & Paz, A. (2013). Metodología para implementar e modelo integrado de Capacidad de Madurez en grupos pequeños y emergentes, 29(122), 177-188.
 - [16] Institute, S. E. (2013). CMMI. Retrieved from <http://www.sei.cmu.edu/cmmi/>
 - [17] Arboleda, H., & Paz, A. (2013). Metodología para implementar e modelo integrado de Capacidad de Madurez en grupos pequeños y emergentes, 29(122), 177-188.
 - [18] Tom DeMarco, T. L. (2012). CMMI in Mexico and around the world (2012). Retrieved from <https://everac99.wordpress.com/tag/cmmi/>
 - [19] Rodríguez-García, J. I. (2010). Normas de Calidad de Software. Retrieved from <http://endrino.pntic.mec.es/jhem0027/calidad/normalizacion/normalizacion.html>