

**CALIDAD DEL SOFTWARE: CAMINO HACIA UNA VERDADERA
INDUSTRIA DE SOFTWARE**
SAULO ERNESTO ROJAS SALAMANCA*
JUAN JOSÉ BORJA PARRA**

RESUMEN

El software es quizá uno de los productos de la ingeniería que más ha evolucionado en muy poco tiempo, pasando desde el software empírico o artesanal hasta llegar al software desarrollado bajo los principios y herramientas de la ingeniería del software.

Sin embargo, dentro de estos cambios, las personas encargadas de la elaboración del software se han enfrentado a problemas muy comunes: unos debido a la exigencia cada vez mayor en la capacidad de resultados del software, debido al permanente cambio de condiciones lo que aumenta su complejidad y obsolescencia; y otros, debido a la carencia de herramientas adecuadas y estándares de tipo organizacional encaminados al mejoramiento de los procesos en el desarrollo del software. Hacia la búsqueda de mecanismos de solución de estos últimos problemas se orienta este artículo.

Una necesidad sentida en nuestro medio es el hecho de que los productos de software deben ser desarrollados con base en la implementación de estándares mundiales, modelos, sistemas métricos, capacitación del recurso humano y otros principios y técnicas de la ingeniería del software que garanticen la producción de software de calidad y competitividad a nivel local e internacional. A su vez, las empresas dedicadas al desarrollo de software pueden certificar sus productos bajo una norma técnica o un modelo de madurez de reconocimiento mundial.

Estos aspectos suponen un reto de investigación permanente en la búsqueda de métricas del software, metodologías para la implementación de normas técnicas, modelos de madurez y estándares mundiales para el aseguramiento de la calidad del software que sirvan para la evaluación y el mejoramiento de los procesos de su desarrollo en nuestro medio. Esta temática también es analizada en este artículo.

* Ingeniero de Sistemas, Universidad Distrital Francisco José de Caldas.
Licenciado en Química, Universidad Nacional de Colombia.
Especialista en Ingeniería de Software, Universidad Industrial de Santander.
Miembro del Comité Técnico de Sistemas de Procesamiento de Información de la
Secretaría Técnica de Normalización de Tecnología Informática.
Docente Investigador, Escuela de Administración de Negocios EAN.
E-mail: eanet1@andinet.com

** Ingeniero de Sistemas, Universidad Distrital Francisco José de Caldas.
Especialista en Ingeniería de Software, Universidad Industrial de Santander.
Docente Investigador, Escuela de Administración de Negocios EAN.
E-mail: eanet1@andinet.com

Antecedentes: Génesis y Evolución

Con el acelerado avance tecnológico de la información, la cantidad y complejidad de los productos de software se están incrementando considerablemente, así como también la exigencia en su funcionalidad y confiabilidad; es por esto que la calidad y la productividad se están constituyendo en las grandes preocupaciones tanto para gestores como para desarrolladores de software.

En los primeros años del software, las actividades de elaboración de programas eran realizadas por una sola persona utilizando lenguajes de bajo nivel y ajustándose a un computador en especial, que generaban programas difíciles de entender, aún hasta para su creador, después de algún tiempo de haberlo producido. Esto implicaba tener que repetir el mismo proceso para desarrollar el mismo programa para otras máquinas. Por consiguiente la confiabilidad, facilidad de mantenimiento y cumplimiento no se garantizaban y la productividad era muy baja.

Posteriormente, con la aparición de técnicas estructuradas y con base en las experiencias de los programadores se mejoró la productividad del software. Sin embargo, este software seguía teniendo fallas, como por ejemplo: documentación inadecuada, dificultad para su correcto funcionamiento, y por supuesto, hasta insatisfacción del cliente.

Conforme se incrementaba la tecnología de los computadores, también crecía la demanda de los productos de software, pero mucho más lentamente, tanto que hacia 1990 se decía que las posibilidades del software estaban retrasadas respecto a las del hardware en un mínimo de dos generaciones de procesadores y que la distancia continuaba aumentando¹. La implementación del enfoque y la programación orientada a objetos en esta década ayudó a disminuir la complejidad de las aplicaciones y a hacerlas más útiles y atractivas.

En la actualidad muchos de estos problemas subsisten en el desarrollo de software, con una dificultad adicional relacionada con la incapacidad para satisfacer totalmente la gran demanda y exigencias por parte de los clientes.

En Colombia, desde hace mucho tiempo se viene desarrollando software y no cabe duda que existe gran capacidad de desarrollo y que sus creadores han logrado mantenerse al día, a pesar de los enormes cambios de la tecnología informática. Sin embargo, este desarrollo no corresponde a una verdadera industria del software, debido a que este proceso ha sido de tipo espontáneo por quienes trabajan en ésta área².

Para ayudar a consolidar un ambiente favorable a la creación de una verdadera industria de este sector, mediante la gestión de calidad, productividad y exportación, es indispensable cumplir con los requisitos que exige el proceso de Normalización en Tecnología de la Informática, el cual se hace en nuestro país a través del ICONTEC (Instituto Colombiano de Normas Técnicas y Certificación) y a nivel mundial la ISO (International Organization for Standardization), que viene estableciendo una serie de normas para el desarrollo y aseguramiento de la calidad del software.

También, Universidades y organizaciones de reconocimiento mundial, tales como: el Software Engineering Institute (SEI) de la Universidad de Carnegie Mellon; el European Software Institute (ESI), La Bell-Core de Canadá, han hecho grandes progresos y aportes creando herramientas y modelos para el mejoramiento de la calidad del software.

Problemática del Desarrollo del Software

El software es quizá uno de los productos de la ingeniería que más ha evolucionado en muy poco tiempo, pasando desde el software empírico o artesanal hasta llegar al software desarrollado bajo los principios y herramientas de la ingeniería del software. Sin embargo, dentro de esta evolución, las personas encargadas de la elaboración del software se han enfrentado a problemas muy comunes, que actualmente se pueden clasificar en dos categorías:

¹ WINBLAND L., Ann. Software Orientado a Objetos. Addison-Wesley. Wilmington USA, 1993.

² DÁVILA S., Manuel. Hacia una Industria de Software en Colombia. En revista Sistemas, Asociación Colombiana de Ingenieros de Sistemas ACIS, No. 72, Julio-Septiembre de 1997. p.14.

1. Incremento en la exigencia de la capacidad de resultados del software y en los permanentes cambios de condiciones para su funcionamiento, lo cual aumenta la complejidad para su elaboración y acelera su obsolescencia. Estos factores hacen más complicado el dimensionamiento de proyectos porque generan incremento en los niveles de esfuerzo para obtener el producto, y ocasionan sobrecostos y retrasos en los mismos.

2. Las empresas y clientes en general están exigiendo la regulación de los productos de software, con base en la implementación de estándares universales encaminados a mejorar los procesos de desarrollo y a asegurar su calidad. Hoy en día las normas técnicas y los modelos para determinar la capacidad de desarrollo de software son requisitos de la contratación para muchas organizaciones, gobiernos o clientes en general; su no implementación implica que el producto obtenido no cumpla con los parámetros de calidad y por consiguiente no sea competitivo tanto nacional como internacionalmente.

Además de esta problemática, caracterizada en el ámbito mundial, en Colombia, existen otros aspectos llamativos; según varios estudios realizados [Cifuentes95], [Cifuentes96], [ACUC95], [ACUC96], los problemas más notorios en el desarrollo de software en Colombia son: la piratería, el desconocimiento del tema por parte de quienes toman las decisiones, la saturación del mercado en algunos segmentos, la falta de capacidad técnica del personal de bajo nivel empresarial, el costo de las herramientas de desarrollo y los cambios tecnológicos³. También son alarmantes las experiencias desastrosas en proyectos de desarrollo de software: sobrecostos, grandes retrasos, falta de calidad, etc⁴.

A pesar de que en Colombia existe gran capacidad de desarrollo y de que los programadores han tratado de involucrar en sus procesos de producción de software los avances de la tecnología informática, la realidad es que las organizaciones y el software producido aún no son competitivos y no se ajustan a estándares mundiales, en normas técnicas o modelos de Madurez tales como: la guía ISO 9000-3, o el CMM (Capability Maturity Model) del Software Engineering Institute (SEI).

Ante este panorama surge una inquietud que debe tener una respuesta satisfactoria en los próximos años y es: ¿Cómo desarrollar en Colombia software competitivo?

Para dar respuesta a este interrogante y solución a la problemática presentada, están apareciendo técnicas y herramientas dentro de la "Ingeniería del Software", que al aplicarlas rigurosamente garantizan: mejoramiento en los procesos de desarrollo de software, calidad en los productos de software y certificación bajo un estándar internacional. Sin embargo, se cuestiona la subjetividad de algunas técnicas y modelos propuestos. Por lo tanto, frente a los modelos recientes se ha generado gran expectativa para su aplicabilidad y desconcierto al determinar cuál es el más conveniente.

Estas dificultades suponen un reto de investigación permanente, en la búsqueda de mecanismos de solución, con la cooperación de las Instituciones universitarias, gremios y proveedores de Software, para crear mecanismos de solución, aprovechando el hecho que cada día es más evidente la necesidad de desarrollar productos de software de calidad con base en la implementación de estándares mundiales, modelos, sistemas métricos, capacitación del recurso humano y otros principios y técnicas de la ingeniería del software. Además, porque algunas empresas persiguen ser certificadas bajo una norma técnica o un modelo de madurez de reconocimiento mundial para comercializar sus productos.

CALIDAD

El uso más frecuente del término calidad cubre el conjunto de atributos o características de un producto, proceso o servicio que suple las necesidades de los clientes; o sea que este término es usado para reflejar la integridad y la utilidad de un producto, la eficiencia y el servicio eficaz.

³ ACOSOFT- Centro Nacional de Consultoría. Estudio de Mercado de Software en Colombia, 1993.

⁴ CIFUENTES, Iván. Estado del Proceso de Desarrollo de Software en Colombia. Memo de Investigación, Marzo de 1996

Una definición apropiada puede ser: "La totalidad de peculiaridades y características de un producto o servicio que determinan su capacidad de satisfacer necesidades declaradas o implícitas"⁵.

La implementación de procesos de mejoramiento de la calidad implica muchas veces cambios en la organización, en la cultura de los analistas y de los directivos. Las expectativas de los beneficios obtenidos deben corresponder a las inversiones y esfuerzos realizados.

CALIDAD DEL SOFTWARE

Al hablar de calidad de software, se debe partir de la definición de software dada por la Norma ISO 9000-3: "Creación intelectual que comprende los programas, los procedimientos, las reglas y cualquier documentación asociada perteneciente a la operación de un sistema de procesamiento de datos". Realmente son cuatro los elementos que conforman el software: programas, procedimientos, reglas y documentación; no debe entenderse como sinónimo de programas o código fuente.

La calidad del Software es normalmente cuestionada porque todavía subsisten principios muy subjetivos para medirla, "es de buena calidad en tanto satisfaga las necesidades de un usuario"; sin embargo involucra otros aspectos. Veamos dos definiciones; Calidad de Software es:

➤ "La concordancia de los requerimientos 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"⁶.

➤ "Consistencia con el diseño, independencia de plataforma y optimización de recursos"⁷.

La calidad del software debe propender para que el software haga lo que se espera de él, dentro del tiempo y presupuesto establecidos y que permita a las partes involucradas alcanzar los objetivos derivados del negocio; por una parte su comercialización apropiada y por la otra, su uso dentro del sistema de información como soporte de sus operaciones.

La calidad del software debe ser el resultado de un proceso sistemático de pruebas, revisiones, detección y corrección de errores, con la funcionalidad adecuada a las necesidades de los usuarios cada vez más exigentes, con características sólidas sobre la fiabilidad y garantía requeridas.

Factores de Calidad del Software

Los productos de software deben tener implícitos una serie de factores de calidad que garanticen la plena satisfacción del cliente y de la empresa desarrolladora tales como: corrección, confiabilidad, eficiencia, integridad, entre otros. Algunas definiciones son las siguientes:

Corrección:

Grado en que un programa satisface sus especificaciones y consigue cumplir los requerimientos del usuario

Confiabilidad:

Probabilidad que programa opere libre de fallos por determinado periodo de tiempo en un medio, previniendo que se use dentro de los límites de su diseño.

Eficiencia:

Grado en el que un software o componente desempeña sus funciones con el consumo mínimo de recursos.

Integridad:

Grado de control del uso y acceso al software por personal no autorizado.

⁵ Norma ISO/IEC 8402. Administración de la Calidad y Aseguramiento de la Calidad. Vocabulario.

⁶ PRESSMAN, Roger S. Ingeniería del Software. Mc Graw Hill, México.

⁷ RUBEY, Ray. Conferencia sobre Aseguramiento de la Calidad del Software SQA.

Facilidad uso:

Esfuerzo requerido para aprender, trabajar, preparar la entrada e interpretación de las salidas del software.

Facilidad de mantenimiento:

Esfuerzo requerido para detectar y corregir un error en un producto de software.

Flexibilidad:

Facilidad con que un software puede modificarse para ser usado en aplicaciones o ambientes diferentes.

Facilidad de prueba:

Esfuerzo requerido para probar el software con el fin de asegurar que realiza las funciones para las cuales fue construido.

Portabilidad:

Facilidad con que un software puede moverse de un ambiente a otro.

Reusabilidad:

Grado en que módulos de un producto de software puede ser utilizado para construir otras nuevas aplicaciones.

Estabilidad:

Medida de la carencia de cambios perceptibles en un producto de software a pesar de alguna ocurrencia en el ambiente del sistema.

Factores en contra de la Calidad del Software⁸

- No existe compromiso gerencial real con los procesos de calidad; éste significa participación, interés y responsabilidad por los resultados obtenidos.

- No existe claridad y conciencia sobre qué es calidad. El hecho de tener la mas variada gama de interpretaciones sobre calidad genera la más variada gama de interpretaciones en cuanto a

las responsabilidades, actitudes y acciones a adelantarse en busca de la calidad.

- No se entrena ni capacita para la calidad. La capacitación, motivación y entrenamiento hacia la calidad, deben conjugar no solo la capacitación en el área técnica de la calidad, sino también en las áreas de la motivación y concientización de lo que es un proceso de mejoramiento de las herramientas específicas para lograrlo.

- Las consecuencias de la falta de calidad no se divulgan con datos. Esto impide que los empleados tengan una idea de los perjuicios reales que sufre la empresa a causa de rechazos, devoluciones y pérdida de imagen en el mercado. El análisis de estos datos con la participación de todos los integrantes, permite obtener una visión adecuada de los problemas existentes y cambiar de actitud, pues la falta de calidad de un producto es el reflejo de quien lo produjo.

- El control de la calidad es visto como una actividad policial. En el pasado las actividades desarrolladas en este campo fueron las de inspeccionar, controlar y rechazar. Hoy en día, desafortunadamente, algunas compañías continúan con este pensamiento obsoleto. Por lo tanto es natural que los empleados tengan un comportamiento negativo frente a quien rechaza su trabajo y dice que no esta bien hecho, sin tener ninguna otra actitud orientadora que lo pueda mejorar y en consecuencia para ser aprobado en el futuro.

ASEGURAMIENTO DE LA CALIDAD DEL SOFTWARE -SQA-⁹

El Aseguramiento de la Calidad se define como: "Toda actividad planeada y sistemática implementada dentro del sistema de calidad y demostrada según sea necesario, para proveer la seguridad adecuada de que una entidad cumplirá los requisitos de calidad"¹⁰.

⁸ GARZÓN, Héctor V. Curso: Validación y Verificación del Software. Aseguramiento de la Calidad del Software, Universidad Industrial Santander UIS, 1995.

⁹ SQA - Software Quality Assurance.

¹⁰ Norma ISO 8402. Administración de la Calidad y Aseguramiento de la Calidad. Vocabulario.

La definición plantea claramente que el Aseguramiento de la Calidad de Software es un proceso que implica planeación, seguimiento y verificación aplicado durante todo el proceso de desarrollo; garantizando al cliente que la organización dispone de los procedimientos efectivos para asegurar la satisfacción de los requerimientos establecidos. Por consiguiente se debe determinar la organización de acuerdo con el sistema de calidad propuesto, con las normas, el sistema de métricas y los procedimientos a implementar. En este sentido, la Norma ISO 9000-3 aporta los elementos que componen un sistema de calidad en el desarrollo de Software y se complementa con el grupo de normas IEEE y NTC sobre calidad.

El Aseguramiento de la Calidad del Software tiene dos dimensiones complementarias: una orientada hacia el producto que hace que éste esté libre de defectos y satisfaga las necesidades de los clientes; y otra orientada hacia el proceso que implica la implantación de normas, técnicas y herramientas durante el ciclo de vida de desarrollo del producto de software.

Proceso de Aseguramiento de la Calidad del Software

Este proceso debe empezar por definir el propósito de la actividad de aseguramiento y las actividades, crear el itinerario, asignar las personas que van a participar, planear y elaborar los documentos de referencia del proyecto, ejecutar y administrar el plan; y describir normas, prácticas y métricas que se usarán teniendo en cuenta las normas, estándares y requerimientos del usuario.

Elementos del Proceso de Aseguramiento de Calidad del Software:

1. Función y contenido del plan de aseguramiento de calidad.
2. Organización para implementar un efectivo SQA.
3. Evaluación de los procesos y productos.
4. Modelos, métricas y herramientas que apoyan el SQA.

1. Función y contenido del Plan de Aseguramiento de Calidad.

El Plan de Calidad es "un documento que enuncia las prácticas, los recursos y la secuencia de actividades relacionadas con la calidad, que son específicas a un producto, un proyecto o un contrato en particular"¹¹. Comprende el manual respectivo junto con los procedimientos detallados que cubren todas las actividades o funciones de una organización respecto a la calidad.

Un plan de aseguramiento de Calidad, para un plan de informática, desarrolla¹²:

- Estructura del Sistema de calidad: Definición de la organización, recurso, responsabilidades del proyecto de Software que se llevará a cabo.
- Actividades del ciclo de vida del proyecto informática: Define fases del proyecto, productos resultantes y acciones de aseguramiento de calidad a aplicar durante el desarrollo del proyecto.

El plan de calidad debe especificar o hacer referencia a los siguientes aspectos¹³:

- Objetivos de calidad, expresados en términos mensurables siempre que sea posible.
- Criterios definidos en cuanto a la entrada y salida para cada fase de desarrollo.
- Tipos de ensayo, verificación y las actividades de validación que se deben efectuar.
- Planificación detallada del ensayo, la verificación y las actividades de validación que se deban efectuar, incluyendo los programas, los recursos y las autoridades de aprobación.

¹¹ Norma ISO 8402. Administración de la Calidad y Aseguramiento de la Calidad. Vocabulario.

¹² QUERUBÍN. Op. Cit.

¹³ Norma NTC-ISO 9000-3. Normas para la administración de la Calidad y aseguramiento de la Calidad. parte 3: Directrices para la aplicación de ISO 9001CEPAL, Comisión Económica para América Latina y el Caribe. Reestructuración industrial y cambio tecnológico: Consecuencias para América Latina. Naciones Unidas. Santiago de Chile, 1989. pp.14-29.

➤ Responsabilidades específicas en las actividades de calidad, tales como: Revisiones y ensayos, administración de la configuración y control del cambio, control de defectos y acción correctiva.

2. Organización para un efectivo SQA

Para un efectivo Aseguramiento de la Calidad del Software la organización debe disponer como mínimo de dos equipos: *Equipo de desarrollo*, que tiene la responsabilidad de establecer los requerimientos, realizar el diseño, escribir el código, realizar las pruebas y preparar los documentos y el de aseguramiento de calidad; y *Equipo de Aseguramiento de Calidad*, que tiene la responsabilidad de revisar que cada una de las fases se realice de acuerdo a lo planeado y asegurarse y en forma correcta.

Se deben tener en cuenta estos puntos importantes:

- Garantizar que los errores encontrados serán corregidos.
- Tener un control directo sobre los conflictos por parte de la dirección.
- Contar con un gerente muy responsable, con mucho interés y compromiso en el proyecto; y que observe y apoye la función del aseguramiento de la calidad del Software.

Debido a su gran responsabilidad, el equipo de aseguramiento debe estar formado por personas calificadas, con amplia experiencia en el desarrollo del software, familiaridad con la aplicación y necesidades del usuario, entendimiento del desarrollo aplicable del software y con las normas de aseguramiento de calidad del software; también deben ser diestros en la comunicación oral y escrita.

3. Evaluación de los procesos y productos

La evaluación de los procesos y de los productos se debe hacer paralelo con el desarrollo del software, verificando que los documentos generados en cada fase concuerdan con los requerimientos del sistema, del software, de diseño, de programación, de pruebas. Esta evaluación im-

plica la realización de revisiones y auditorías formales, verificando que los problemas detectados sean corregidos. Los cambios suscitados se deben documentar y administrar.

4. Modelos, métricas y herramientas que apoyan el SQA

Existen en la actualidad diferentes modelos, unos encaminados a la detección y corrección de errores, y otros orientados a la evaluación de los procesos del software con el objetivo de mejorarlos. Estos modelos contemplan tanto los aspectos cualitativos, como los aspectos cuantitativos (o de medición) de los procesos del desarrollo de software. De hecho las métricas hacen referencia a la parte cuantitativa de la calidad. Cada uno de estos modelos se apoya en herramientas para disminuir el número de errores y defectos y mejorar significativamente los procesos y la evaluación del software.

En el siguiente punto se realizará un somero análisis sobre estos modelos, métricas y herramientas, junto con una descripción breve sobre normas técnicas y otros aspectos que apoyan el aseguramiento de la calidad del software.

ENFOQUES PARA EL MEJORAMIENTO DE LOS PROCESOS DE DESARROLLO Y DE LA CALIDAD DEL SOFTWARE

Las organizaciones preocupadas por mejorar sus procesos de desarrollo y por hacer sus productos altamente competitivos, se han orientado unas hacia la implementación de modelos desarrollados por organizaciones de reconocimiento mundial tales como el CMM (Capability Maturity Model) del SEI (Software Engineering Institute), estándar SPICE (Software Process Improvement and Capability dEtermination) de la ISO/IEC (International Organization for Standardization / International Electrotechnical Commission) o el modelo TRILLIUM de BELLCORE (Canadá), entre otros; otras, por implementar estándares mundiales como las normas técnicas; por ejemplo la serie ISO 9000-3 de ISO/IEC, o el Std.810 de la IEEE (Institute of Electrical and Electronic Engineers).

Con el surgimiento de diferentes modelos, normas, métodos y herramientas para evaluar y administrar los procesos de software, se generó al-

gún desconcierto en los desarrolladores porque no sabían cómo determinar el mejor y/o cómo aplicarlo de acuerdo con sus necesidades. Es por esto que los aspectos relacionados con calidad y aseguramiento de la calidad del software, tales como: métricas, pruebas y herramientas de calidad de software, modelos de estimación de costos en proyectos de software, normas técnicas, CMM, estándar SPICE y modelos de prevención y detección de errores¹⁴ deben ser conocidos por las organizaciones con el fin de considerar su implementación o qué pasos seguir, para lograr óptimos resultados en sus productos de software y de apoyo a las metodologías y técnicas de desarrollo definidas.

METRICAS DEL SOFTWARE

La medición del software es una estrategia esencial en el esfuerzo de conocer y clarificar los problemas que atentan contra la confiabilidad y desarrollo de productos de calidad. Como bien lo afirma Lord Kelvin: "Cuando puedes medir lo que estás diciendo, y expresarlo en números, sabes algo sobre ello, pero cuando no puedes medirlo, cuando no puedes expresarlo con números, tu conocimiento es escaso e insatisfactorio: puede ser el comienzo del conocimiento, pero en tus pensamientos, apenas estás avanzando hacia el escenario de la ciencia"¹⁵.

En este sentido, la medición del software se está convirtiendo en parte integral para mejorar el proceso y el producto mismo, ya que a partir de la implementación de un sistema de métricas adecuado se logra incrementar los niveles de calidad y productividad en las organizaciones. Sin embargo, la realidad indica que la gran mayoría de proyectos de software no lo hacen, debido muchas veces al desconocimiento de las métricas, a la carencia de técnicas a los costos y riesgos generados en su implementación.

¿Qué es una Métrica de software ?

Se puede decir que una *Métrica de Software* es un conjunto de unidades con base en las cuales es posible medir un atributo cuantificable del software dentro de una escala de valores predeterminada.

Análisis de Métricas del Software se refiere a la disciplina de recoger y analizar datos de soft-

ware y a las mediciones reales de los atributos del software, considerados éstos como características puntuales del producto o del proceso de desarrollo de software que permite obtener información útil acerca del estado del producto o del progreso del proyecto.

Es necesario medir el software por las siguientes razones:

- **Entender el proceso de producción;** que a su vez permite:

1. Determinar qué herramientas utilizar, que ayuden al mejoramiento y optimización de los procesos de desarrollo.
2. Establecer estrategias más efectivas en la gestión y planeamiento del proceso de producción.
3. Determinar en qué niveles de productividad se encuentran los procesos de desarrollo.

- **Mejorar el producto de Software,** incrementar los índices de calidad de ciertos factores.

En general, el software como cualquier producto de la ingeniería, debe ser medido porque es necesario predecir y supervisar lo que se hace y cómo se hace, con el propósito de minimizar costos de producción y optimizar los procesos de desarrollo en la búsqueda de una mayor calidad, productividad y competitividad.

Qué se debe medir en el software ?

Existe cierta dificultad para medir el software, porque es conocimiento, lo cual marca una gran diferencia con la medición que se realiza sobre otros productos de la Ingeniería y de la ciencia.

Es de anotar que hasta ahora no existe un procedimiento definido que fije las pautas sobre cuáles métricas implementar y la forma de hacerlo.

¹⁴ Estos temas son estudiados como módulos dentro de la investigación "Normalización y Calidad en la Producción de Software" que actualmente se desarrolla en la Escuela de Administración de Negocios EAN, bajo la coordinación de los autores del presente artículo.

¹⁵ PRESSMAN, Roger. Ingeniería de Software, un Enfoque Práctico, Tercera Edición. Mc Graw Hill, 1993.

Dependiendo del tipo de software y en sí de cada proyecto, se deben establecer los procesos de documentación, objetivos, metas y en qué periodos de tiempo se van a obtener para definir las medidas a utilizar y poder calibrar periódicamente el progreso, medir el desempeño del software y por ende tener un control del proyecto en cuestión. Además, estos datos servirán de base para la estimación y corrección de errores y defectos de futuros desarrollos.

Clasificación de las Métricas del Software

En la bibliografía se encuentran diferentes tipos de métricas (productividad, calidad, técnicas, mantenimiento, etc); en este documento se clasifican como:

1. Métricas orientadas al Producto. Están dirigidas hacia la medición del tamaño, complejidad y en general, de los factores de calidad del producto. Ejemplos: Líneas de Código (LDC), complejidad, puntos de función, número de errores después de entrega, mantenimiento del producto, etc.

2. Métricas orientadas al Proceso. Están dirigidas hacia la medición de la productividad, recursos y en general, hacia los aspectos de calidad del proceso de desarrollo de software. Ejemplos: número de personas, calendario requerido, número total de errores cometidos y corregidos, margen de fracaso, métricas de productividad.

Ejemplos de Métricas

A continuación se hace una breve descripción sobre algunas métricas recopiladas dentro de la investigación y que en consideración a la documentación analizada involucran aspectos de gran trascendencia en el desarrollo del software.

- Métricas para la calidad del software

McCall propone una serie de métricas basadas en la medición de factores de calidad de software, tal como se presenta en la Tabla No.1.

- Otras métricas.

En la Tabla No.2 aparece el nombre, la definición y las unidades de medida propuestas para cada métrica.

NORMALIZACIÓN Y CERTIFICACIÓN DEL SOFTWARE

Tanto el comercio como la industria en el ámbito mundial tienden a adoptar reglas de producción y comercialización uniformes. Es decir, tienden hacia la "Estandarización" mediante el uso de normas; lo cual se traduce no solo en la reglamentación de la producción de bienes y servicios, sino que va más allá, ya que propende por asegurar la economía, ahorrar gastos y garantizar el funcionamiento rentable de las empresas.

¿Qué es una Norma Técnica ?

El Decreto 2269¹⁶ define una *Norma Técnica* como: "Documento establecido por consenso y aprobado por un organismo reconocido, que suministra, para uso común y repetido, reglas, directrices y características para las actividades o sus resultados, encaminados al grado óptimo de orden en un contexto dado. Las normas técnicas se deben basar en los resultados consolidados de la ciencia, la tecnología y la experiencia, y sus objetivos deben ser los beneficios óptimos para la comunidad".

¿Por qué utilizar Normas ?

Como parte de su estrategia global, la mayoría de las empresas han abrazado las normas internacionales como una herramienta clave para abrir mercados; este es el panorama que se observa actualmente, dado que la actual recesión económica obliga a las empresas a ocuparse no solo de su mercado local, sino mirar hacia mercados externos, como una oportunidad para su supervivencia.

¹⁶ Decreto 2269. Organización del Sistema Nacional de Normalización, Certificación y Metrología. Ministerio de Desarrollo Económico, república de Colombia. 16 de Noviembre de 1993.

TABLA No.1

ASPECTO	FACTOR DE CALIDAD	DEFINICIÓN
Operación del Producto	Corrección	Grado en que un programa satisface sus especificaciones y consigue los objetivos de la misión encomendada por el cliente.
	Fiabilidad	Grado en que se puede esperar que un programa lleve a cabo sus funciones esperadas con la precisión requerida.
	Eficiencia	Cantidad de recursos de hardware y de código requerido por un programa para realizar su función.
	Integridad	Grado en que puede controlarse el acceso al software o a los datos por personas no autorizadas.
	Facilidad de uso	Esfuerzo requerido para aprender, trabajar y preparar la entrada e interpretar la salida de los programas.
Revisión del Producto	Facilidad de mantenimiento	Esfuerzo requerido para localizar y arreglar un error en un programa.
	Flexibilidad	Esfuerzo requerido para modificar un programa operativo
	Facilidad de prueba	Esfuerzo requerido para probar un programa en forma que se asegure que realiza la función querida.
Transacción del Producto	Portabilidad	Esfuerzo requerido para transferir el programa desde una configuración de hardware o sistema operativo a otro.
	Reusabilidad	Grado en que un programa (o partes de él) se puede reutilizar en otras aplicaciones.
	Interoperabilidad	Esfuerzo requerido para acoplar un sistema a otro.

Las normas internacionales pueden ser usadas estratégicamente por las organizaciones, pues permiten:

1. Fortalecer el mercado nacional.
2. Eliminar restricciones al comercio.
3. Unificar mercados y requisitos técnicos, incrementando las oportunidades comerciales.
4. Producir y comercializar el mismo producto mundialmente.
5. Fijar pautas para la entrada a nuevos mercados.

Se puede afirmar que los estándares o normas desempeñan un papel fundamental en el intercambio comercial entre los países y dentro de cada uno de ellos, lo cual obliga a su análisis para adaptarlas a las necesidades del entorno. Respecto a la Calidad del Software y al Asegura-

miento de la Calidad del Software, cabe anotar que en este momento, en Colombia se están revisando las normas internacionales sobre sistemas de calidad existentes para adaptarlas como Normas Técnicas Colombianas dentro del Comité Técnico de Sistemas de Procesamiento de Información (ICONTEC), con el fin de garantizar en el ámbito nacional, los intereses del sector de la producción de software.

Marco Legal

Las normas internacionales sobre sistemas de administración de calidad tienen sus antecedentes en normas militares y del sector nuclear que datan de más de 20 años y que se han extendido a lo largo de los años a otros sectores como el químico, el farmacéutico, el software, etc.

El Comité Técnico de la ISO TC 176 "Administración y Aseguramiento de la Calidad" preparó desde 1979 hasta emitir en 1987 las primeras normas básicas relativas a los sistemas de Adminis-

tración de la Calidad¹⁷, la serie ISO 9000; éstas normas han sido objeto de revisión, y sus últimas versiones publicadas en julio de 1994 y 1997. Además de la ISO, existen otras entidades dedicadas a la elaborar estándares sobre Aseguramiento de la Calidad tal como el IEEE, con su norma IEEE Std. 730.1 de 1989¹⁸ y otras. Otras normas de gran importancia, son las Australia-
nas.

Estas normas fueron precedidas por la de terminología publicada en 1986, ISO 8402 "Administración de la Calidad y Aseguramiento de la Calidad. Vocabulario" que presentó la terminología básica de administración y aseguramiento de la calidad.

En Colombia la Superintendencia de Industria y Comercio es el organismo encargado de: establecer, coordinar, dirigir y vigilar los programas nacionales de control industrial de la calidad, medidas y metrología, así como de organizar y adecuar los laboratorios que considere indispensables para cumplir a cabalidad con sus funciones. Los estándares para la normalización y metrología en Colombia, están basados en:

- Decreto No.2269 del 16 de Noviembre de 1993: Organiza el sistema Nacional de Normalización, Certificación y Metrología.

- Resolución 140 del 4 de Febrero de 1994: Establece el procedimiento para la acreditación; regula actividades dentro del Sistema Nacional de Normalización, Certificación y Metrología.

- Decisión 376, Sistema Andino de Normalización, Acreditación, Ensayos, Certificación, Reglamentos Técnicos y Metrología.

El proceso de Normalización de Tecnología Informática se hace a través del ICONTEC (Instituto Colombiano de Normas Técnicas y Certificación), quién para cumplir con este propósito realizó un convenio con la ACUC¹⁹ creando la Secretaría Técnica de Normalización de Tecnología Informática en 1996; cuyo trabajo se ha centrado en el análisis, revisión y adopción de Normas ISO relacionadas con la Ingeniería del software como Normas Técnicas Colombianas (NTC) y en la divulgación de esta normas.

Definiciones relacionadas con los aspectos sobre Normalización.

➤ Normalización: "Es la actividad que establece, en relación con problemas actuales o potenciales, soluciones para aplicaciones repetitivas y comunes, con el objeto de lograr un grado óptimo de orden en un contexto dado. En particular consiste en la elaboración, la adopción y la publicación de las normas técnicas"²⁰.

➤ Certificación: "Procedimiento mediante el cual una tercera parte da constancia por escrito o por medio de un sello de conformidad de que un producto, un proceso o un servicio cumple los requisitos especificados en el reglamento"²¹. En Colombia, la Certificación es un documento emitido por ICONTEC, por el cual se certifica que un producto asegura y garantiza la satisfacción del consumidor hacia productos que puede adquirir, esta satisfacción del cliente es dada por el seguimiento y cumplimiento de las normas y estándares, o por un trato entre cliente y proveedor.

➤ Acreditación: "Procedimiento mediante el cual se reconoce la competencia técnica y la idoneidad de organismos de certificación e inspección, laboratorios de ensayo y metrología para que lleven a cabo sus actividades"²².

Normas Técnicas sobre Aseguramiento de la Calidad del Software

- Norma Técnica Colombiana NTC – Serie ISO 9000: Normas orientadas hacia la gestión y Aseguramiento de la Calidad. Establece lineamientos sobre los principales conceptos relativos a la ca-

¹⁷ Catálogo (NTC) 1996. Descripción del procedimiento de Certificación y Aseguramiento de Calidad.

¹⁸ Norma IEEE 730-89. Estándar de Planes de Garantía para el Desarrollo de Software de Calidad.

¹⁹ Asociación Colombiana de Usuarios de Informática y Comunicaciones.

²⁰ Decreto No. 2269. organización del sistema Nacional de Normalización, Certificación y Metrología. 16 de Noviembre de 1993.

²¹ Decreto No. 2269. Op. Cit.

²² Ibid.

TABLA No.2

METRICA	DEFINICIÓN	UNIDADES DE MEDIDA
Control del Progreso	Establece el promedio del progreso en el desarrollo de un proyecto determinado.	Meses, años, días de la estimación del proyecto
Equipo de apoyo de software	Identifica el número de Ingenieros del software, personal de primera línea involucrado en el sistema	Número de personas incluidas en el proyecto
Programación del servicio de respuesta	Determina el tiempo transcurrido desde el momento en que se solicita el servicio de respuesta al problema hasta que culmina.	Tiempo hombres-mes
Instancias ADA	La distribución de fallas conduce hacia la naturaleza de la carga de trabajo con el mantenimiento	No. de fallas estimadas No. Correcciones
Distribución Tipo de Fallas	Permite determinar qué tipo de falla está siendo corregida	Número de fallas. Número de correcciones
Análisis de Fallas	Permite encontrar y remover posibles errores en el producto, a partir del análisis del origen de la falla, el tipo y el modo de operar de las fallas.	Número de fallas Tipo de falla
Defectos	Permite determinar el número de defectos por fase (Defecto: imperfección de un producto o proceso de software)	Número de defectos encontrados
Líneas de Código	Permite cuantificar el tamaño del producto por la cantidad de líneas de código fuente que tiene un programa	Líneas de Código (LDC) o Miles de LDC (KLDC)
Puntos de Función	Cuantifica el tamaño del producto mediante el análisis de la funcionalidad del software, utilizando fórmulas empíricas. Aunque esta métrica tiene un grado de subjetividad en la valoración de la complejidad, ha sido una de las más usadas a nivel mundial.	Puntos de Función (PF)
Complejidad Ciclomática	Número clásico ciclomático de la teoría de grafos, indicando el número de regiones en la gráfica. Aplicado al software, es el número de caminos linealmente independientes que componen un programa, por medio del cual se puede establecer la cantidad de pruebas a realizarse.	Caminos linealmente independientes de un programa. Calculados por: $M = V(G) = e - n + 2p$
Tiempo de Duración del Proyecto	Tiempo medido en horas calendario desde el inicio del proyecto hasta la distribución al primer consumidor.	Meses por fase
Tiempo Estimado de Duración	Tiempo de duración del proyecto estimado en horas calendario.	Meses por fase
Movimiento del grupo de trabajo	Determina la cantidad de entrada y salida del recurso humano que se ha realizado a través del proyecto.	% personas retiradas en los últimos 12 meses
Nivel de Maduración	Mide el proceso evolutivo que ha sufrido el proyecto durante todo su desarrollo	Estimación o evaluación
Nivel de Satisfacción	Medida subjetiva que determina el grado de satisfacción del cliente hacia el producto de software	Niveles de satisfacción: bajo, medio, bueno, excelente.
Nivel de entusiasmo	Medida subjetiva que determina el grado de entusiasmo que el grupo muestra hacia el desarrollo del proyecto	Niveles de entusiasmo: bajo, medio, bueno, excelente.
Estructura	Determina las uniones entre los componentes del sistema	LDC
Estabilidad	Determina el grado en que el proyecto se desarrolla establemente cumpliendo los requisitos predefinidos de la calidad y productividad	Número de peticiones de cambio del usuario

lidad y las directrices para elegir y utilizar las normas de sistemas de calidad ISO - ICONTEC, que se pueden usar para propósitos de aseguramiento interno de la gestión de calidad o las que se pueden emplear para asegurar externamente la calidad.

De esta Norma se desprenden cuatro normas, según su orientación:

1. ISO 9001: Sistemas de calidad. Modelo de aseguramiento de la calidad aplicable al diseño, desarrollo, fabricación, instalación y servicio.

2. ISO 9002: Sistemas de calidad. Modelo de aseguramiento de la calidad aplicable a la fabricación e instalación.

3. ISO 9003: Sistemas de calidad. Modelo de aseguramiento de la calidad aplicable a la inspección y ensayos finales.

4. ISO 9004: Gestión de calidad y elementos del sistema de calidad. Lineamientos.

- Norma Técnica Colombiana NTC-ISO 9000-3. Normas para la administración de la calidad y aseguramiento de la calidad Parte 3: directrices para la aplicación de ISO 9001 al desarrollo, suministro y mantenimiento del software. Busca: demostrar la capacidad del proveedor de estos servicios al momento de realizar procesos de contratación; describir los controles y métodos recomendados para producir Software que cumpla con los requerimientos del comprador.

- Norma Técnica Internacional ISO - IEC 12207. Tecnología de la Información. Procesos del Ciclo de Vida del Software. Busca: establecer un marco común para los procesos del Ciclo de Vida del Software (definición, control y mejora); definir la terminología adecuada para la industria del software; servir de apoyo a adquirientes de sistemas, productos y servicios de Software; servir de orientación a desarrolladores, operadores y personal involucrado con los productos de software.

De este grupo de normas las siguientes se estudiaron en el Comité de Tecnología Informática durante 1998 para ser adoptadas como NTC: ISO/IEC 12119, ISO/IEC 13236, ISO/IEC 2382-1, ISO/

IEC 10164-11, ISO/IEC 14143-1.2, ISO/IEC 14598-5. Una vez sometidas a la corrección de estilo en el ICONTEC, estas normas son puestas a discusión pública, para luego ser declaradas como NTC.

MODELOS DE MEJORAMIENTO DE LOS PROCESOS DE SOFTWARE

Existen diversos modelos enfocados hacia el mejoramiento de los procesos de software. Dentro de los más importantes se encuentran:

- Estándar SPICE. Proyecto ISO (International Organization for Standardization).

- CMM, elaborado por el SEI (Software Engineering Institute).

- Bootstrap elaborado por el ESI (European Software Institute).

- Trillium, basado en el CMM desarrollado por Bellcore de Canadá.

En este punto se hará una breve descripción del CMM y del estándar SPICE por ser los de mayor importancia y reconocimiento mundial.

CMM - CAPABILITY MATURITY MODEL

CMM es un modelo propuesto por el SEI (Software Engineering Institute) de la Universidad Carnegie-Mellon de Estados Unidos, que ofrece los medios para evaluar la habilidad de las organizaciones para producir software de buena calidad. Igualmente puede ser usado por:

➤ Un asesor externo para determinar si la organización se puede comprometer en la subcontratación de desarrollos de software.

➤ Las organizaciones para descubrir sus fortalezas y debilidades con respecto a los procesos de software, identificando las áreas en las que debe mejorar.

Para qué sirve CMM ?

Aunque CMM no sirve para resolver a corto plazo los problemas de calidad y de un proyecto atrasado, básicamente permite:

TABLA No. 3
NORMAS TÉCNICAS COLOMBIANAS NTC

Norma NTC	Norma ISO	Nombre	Descripción
2363 (87)	2382-1	Procesamiento de la información. Vocabulario. Parte 1. Términos fundamentales. ?	Establece definiciones, procesos principales, equipos utilizados; representación, organización y presentación de datos; programación y operación de computadores, periféricos y aplicaciones particulares
2439 (88)	2382-2.	Procesamiento de la información. Vocabulario. Parte 2. Operaciones aritméticas y lógicas.	Contiene términos y definiciones
2554 (89)	2382-4	Sistemas de procesamiento de la información. Vocabulario. Parte 4. Organización de datos.	Presenta principios y reglas, términos y definiciones.
2634 (96) Primera Actualiz.	2382-8:1986	Sistemas de procesamiento de la información. Vocabulario. Control, integridad y seguridad.	Contiene principios y reglas, términos y definiciones.
2899 (91)	2832-16	Sistemas de procesamiento de la información. Vocabulario. Teoría de la información.	Contiene términos y definiciones
2900 (91)	2382	Sistemas de procesamiento de la información. Procesamiento de datos distribuido. Vocabulario.	Contiene principios y reglas, términos y definiciones.
2921 (91)	5807	Sistemas de procesamiento de la información. Símbolos de documentación, y convenciones aplicables a los diagramas de flujo de datos, de programación y de sistemas y a los gráficos de redes de programas y recursos del sistema	Contiene: definiciones, diagrama de flujo de datos, diagrama de flujo de programación, diagrama de flujo de sistema, símbolos, convenciones, convenciones especiales.
2951 (91)	2382-19	Sistemas de procesamiento de la información. Vocabulario. Computación análoga.	Contiene generalidades, términos y definiciones.
2968 (96) 1ª Actualiz	2382-9 (1995)	Tecnología de la información. Vocabulario. Comunicación de datos.	Contiene términos y definiciones, medios de transmisión y técnicas de transmisión.
3306 (92)	6592	Sistemas de procesamiento de la información. Documentación. Guías para la documentación de sistemas de aplicación basados en computador	Contiene estudios de factibilidad, especificación del diseño y desarrollo de sistemas, soporte del sistema, implantación del sistema, revisiones después de la implantación, admón. de documentos.
3349 (97) Primera Actualiz.	2382-14.	Sistemas de procesamiento de la información. Vocabulario. Confiabilidad, mantenimiento y disponibilidad.	Define los conceptos básicos que involucran la confiabilidad, el mantenimiento y la disponibilidad en el procesamiento de datos.
3560 (93)		Sistemas de procesamiento de la información. Procedimientos y desarrollo. Guía para la adquisición de sistemas de computación	Establece la recomendación para tener en cuenta en la adquisición de sistemas de computación
3574 (93)	9127	Sistemas de procesamiento de la información. Documentación. Documentación para el usuario e	Contiene definiciones, categorías de la información, documentación del usuario, documentación de referencia,

**TABLA No. 3 (CONTINUACIÓN)
NORMAS TÉCNICAS COLOMBIANAS NTC**

3975 (96)	2382-6	Sistemas de procesamiento de la información. Vocabulario. Preparación y manejo de datos	Contiene principios y reglas, términos y definiciones, ordenamiento, clasificación, confrontación y preparación de los datos.
4094 (97)	2382-12	Sistemas de procesamiento de la información. Vocabulario. Equipo periférico	Define unidades de procesamiento aritméticas, registro y convertidores. Facilita la comunicación internacional en el procesamiento de información. Contiene términos y definición de conceptos seleccionados relevantes al campo del procesamiento de información.
4122 (97)	2382-1	Sistemas de procesamiento de información. Vocabulario. Unidades de procesamiento.	Define unidades de procesamiento, unidades aritméticas, registros y convertidores. Facilita la comunicación internacional en el procesamiento de información. Presenta términos y definiciones de conceptos seleccionados relevantes al campo de procesamiento de información. Contiene principios y reglas.

**TABLA No. 4
NORMAS ISO/IEC e IEEE RELACIONADAS CON EL DESARROLLO DE SOFTWARE**

Norma	Descripción
ISO/IEC 12119	Tecnología de la información. Paquetes de software. Requerimientos de calidad y pruebas
ISO/IEC 2382-1	Tecnología de la información. Vocabulario. Parte 1. Términos fundamentales
ISO/IEC 10164-11	Tecnología de la información. Interconexión de sistemas abiertos. Administración de sistemas objetos y atributos métricos
ISO/IEC 14143-1.2	Tecnología de la información. Medición del software. Medición del tamaño funcional. Parte 1. Definición de conceptos.
ISO/IEC 14598-5	Tecnología de la información. Evaluación del producto de software. Parte 5. Proceso para evaluadores
ISO/IEC 14980	Tecnología de la información. Código de práctica para la administración de la seguridad de la información
ISO/IEC 15026	Tecnología de la información. Niveles de integridad de sistemas y de software
ISO/IEC 2382-17	Tecnología de la información. Vocabulario. Parte 17. Bases de Datos
ISO/IEC 13236	Tecnología de la información. Calidad del servicio. Marco de trabajo.
IEEE 730-1989	Estándar de planes de garantía para el desarrollo de software de calidad
IEEE 1042-1987	Plan de administración de la configuración
IEEE 828-1990	Planes de configuración de administración
IEEE 610.12-1990	Glosario de terminología de ingeniería de software
IEEE 1012-1986	Plan para la verificación y validación del software
IEEE 1074-1995	Estándar para el desarrollo del proceso del ciclo de vida del software
IEEE 1016-1987	Prácticas recomendadas para la descripción del diseño del software
IEEE1008-1987	Cuestionario único para el software
AS 3563.1 -1991	Estándar Australiano para el sistema de administración de la calidad del Software

- Evaluar la habilidad de las organizaciones para producir software de buena calidad.

- Determinar si la empresa puede comprometerse para subcontratación de desarrollos de software.

- Descubrir sus fortalezas y debilidades.

- Ordenar los problemas del proceso de desarrollo.

- Establecer prioridades para ir mejorando esos procesos.

- Ganar control de desarrollo y mantenimiento.

- Mantener un proceso de mejoramiento continuo.

Madurez del Proceso de Desarrollo

En muchas organizaciones los proyectos son excesivamente largos y el presupuesto planeado se incrementa hasta en el cien por ciento, debido frecuentemente a que éstas no poseen infraestructura y soporte necesarios para evitar estos problemas. Algunas de estas organizaciones, que se suelen llamar indisciplinadas, desarrollan proyectos individuales de software que producen resultados excelentes, debido generalmente al esfuerzo heroico de las personas que integran los equipos de trabajo; para repetir este éxito, debe disponerse del mismo equipo que asuma el siguiente proyecto.

Organizaciones Inmaduras.

- Proceso generalmente improvisado (práctica y manejo).

- El Proceso no es rigurosamente seguido o ejecutado, aún si es especificado.

- Organización reaccionaria: resolver crisis inmediatas.

- Agenda y presupuesto que usualmente se exceden.

- Calidad y funcionalidad a menudo comprometidas.

Organizaciones Maduras.

Se caracterizan por:

- Gran habilidad en el manejo de procesos de software.

- Proceso comunicado al personal existente y nuevo.

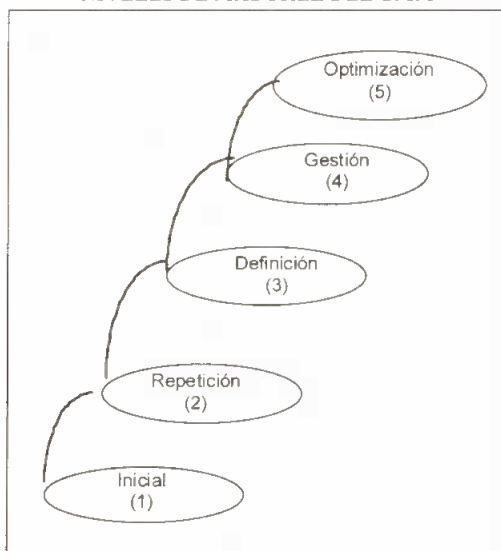
- Actividades ejecutadas según proceso planeado.

- Procesos ajustados a medida que sea necesario (en momentos definidos).

- Procesos definidos que se actualizan cuando es necesario.

- Las mejoras son desarrolladas a través de pruebas controladas y con análisis de costo/beneficio.

GRÁFICA No.1
NIVELES DE MADUREZ DEL CMM



- Reglas y responsabilidades claras en proceso definido.

- Agendas y presupuestos basados en ejecuciones anteriores y sus estimativos son reales.

- Resultados esperados para costos, agenda, calidad y funcionalidad del producto son alcanzados.

- Proceso disciplinado (consistentemente seguido) porque los participantes entienden su valor y cuentan con la infraestructura necesaria para soportarlo.

Estructura de CMM

CMM permite evolucionar desde un proceso caótico, hasta un proceso completamente definido,

predecible y que esté en mejora continua, a través de una serie de prácticas. En la gráfica No.1 se aprecian los Niveles de Madurez que forman la estructura básica de CMM.

ESTÁNDAR SPICE (Software Process Improvement and Capability dEtermination)

Es un Estándar Internacional que provee una estructura para evaluar los procesos de software, que puede ser usada en diferentes ámbitos en una organización donde se requiera efectuar planeación, administración, monitoreo, control, mejoras en la adquisición, desarrollo, operación, evolución y soporte de software.

El estándar SPICE tendrá tres aplicaciones fundamentales: Evaluación de procesos, Mejora de procesos y Determinación de capacidades, y ha sido diseñado para satisfacer las necesidades de clientes, proveedores y asesores como lo muestra la tabla No. 5.

Antecedentes

El surgimiento de diversos modelos para el mejoramiento y evaluación de la capacidad del software, creó desconcierto para determinar cuál aplicar en una organización dada. Para obviar esta situación, se realizó un estudio sobre las necesidades y requerimientos para crear un estándar de evaluación de procesos, realizado por parte del Ministerio de Defensa del Reino Unido y de la ISO

En particular este estudio, arrojó los siguientes resultados:

- Los grandes compradores de software tienen necesidades similares.
- Los desarrolladores de sistemas están interesados en la automejora.

TABLA No.5

GRUPO	BENEFICIOS
Cliente	Habilidad para determinar la capacidad común y potencial de los procesos de software suministrados
Vendedor	<ul style="list-style-type: none"> • Habilidad para determinar la capacidad común y potencial de sus propios procesos de software. • Habilidad para definir áreas y prioridades para mejorar los procesos de software. • Estructura para definir una guía para mejorar los procesos de software.
Asesor	Estructura más organizada para mejorar los aspectos relevantes en la dirección o conducción de un proyecto.

Estos beneficios permitirán: profesionalizar la oferta; cambiar las estrategias de comercialización al nivelar empresas grandes, medianas o pequeñas con base en su capacidad y no en el presupuesto de comercialización o en la imagen corporativa; profesionalizar la demanda con base en el perfil de capacidades; comparar desempeños entre empresas. En términos prácticos esto significa desarrollar un proceso de maduración tanto de la oferta como de la demanda de servicios, abriendo un conjunto de posibilidades para la Industria del software de nuestro país con el propósito de ofrecer productos competitivos.

- Existe una variedad de métodos de evaluación y mejora.
- Existe el deseo de un enfoque común.
- Se busca un énfasis en la Autoevaluación y no en la certificación.

Este estudio dio como resultado la necesidad de elaborar un estándar que: pudiera ser usado para la mejora de procesos y la determinación de capacidades, a partir de un consenso internacional, y que estuviera basado en las mejores y más probadas características de los métodos y

modelos existentes; y que además fuera desarrollado rápidamente.

Después de una serie de consultas internacionales entre los países miembros del SC7²³ en el WG10²⁴, se inicia el proyecto ISO/SPICE para elaborar el estándar internacional, con base en el estudio, participación e integración de los principales modelos, basados en el estándar ISO-9000, como son: TICK-IT (metodología Británica), CMM (del SEI), BOOTSTRAP (metodología europea) y TRILLIUM (modelo canadiense); y financiado principalmente por la industria y algunos gobiernos a través de centros de investigación conjuntos, como el SEI (Software Engineering Institute)²⁵ y el ESI (European Software Institute, ubicado en Bilbao y co-financiado por la Comunidad Económica Europea y el Gobierno Vasco).

ESTRUCTURA DEL SPICE

El Modelo ISO/SPICE es una síntesis de varios de los modelos existentes. Define una serie de conceptos fundamentales para su correcta interpretación:

Proceso: Secuencia de operaciones o prácticas para realizar un propósito.

Proceso de Software: Secuencia para conceptualizar, diseñar, construir, probar, controlar y asegurar la calidad de un sistema o software.

Capacidad del Proceso: Descripción del rango de resultados esperados que pueden lograrse al seguir un determinado proceso.

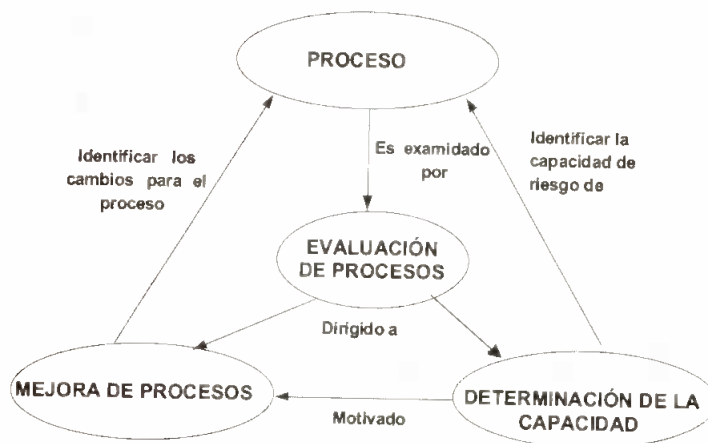
Desempeño del Proceso: Representan los resultados reales que se obtienen al seguir un determinado proceso.

CONCLUSIONES Y RECOMENDACIONES

➤ A través del análisis de la información recopilada, se ha podido determinar que el éxito en los proyectos de software se logra mediante la aplicación de una serie de procesos organizados, que se deben definir con base en: normas técnicas, enfoques de mejoramiento de la calidad, sistema métricos y experiencia obtenida en otros proyectos.

➤ La experiencia y algunos estudios demuestran que en Colombia la mayoría de organizaciones que llevan a cabo proyectos de software lo hacen sin ningún tipo de planeación, ni estándares; situación que cataloga al desarrollo de software como artesanal.

GRÁFICA No. 2
EVALUACIÓN DEL PROCESO DE SOFTWARE



➤ La calidad en la producción de software debe convertirse en una forma de trabajo del Ingeniero de Software, por lo tanto es indispensable que el sector productivo y las universidades propicien y apoyen investigaciones en este sentido y las involucren en los planes de estudio de los programas de Ingeniería de Sistemas.

➤ El estudio de la cuantificación de la calidad del software es muy amplio y existen numerosas métricas definidas pero hay algunas vi-

²³ SC7: Subcomité 7 (Ingeniería de Software) de la ISO.

²⁴ WG10: Grupo de Trabajo 10 del SC7.

²⁵ El SEI es precisamente una de las más importantes recomendaciones del estudio del departamento de la defensa en 1987, se encuentra en la universidad Carnegie Mellon y fue fundado por el Departamento de la Defensa de Estados Unidos.

ciadas por la subjetividad, lo cual incrementa la probabilidad de establecer apreciaciones inadecuadas sobre el grado de calidad de un producto.

➤ Existen buenas normas, modelos y técnicas para crear software de calidad, situación que beneficia a las casas desarrolladoras de software, por ejemplo CMM, norma ISO 9000-3, SPICE son estándares de gran importancia para aquellas empresas que desean evaluar y mejorar la producción del software.

➤ Los estudios sobre métricas del software son muy escasos, especialmente los de complejidad ciclométrica y los pocos que se adelantan, apuntan hacia los mismos aspectos; no se conocen estudios extensos alrededor de métricas de software orientadas a objetos o bases de datos.

BIBLIOGRAFÍA

CIFUENTES, C. Iván. "Una investigación para determinar cómo mejorar el desarrollo de Software en Colombia", Memo de investigación 233, Uniandes. 1996.

DÁVILA S., Manuel. Hacia una Industria de Software en Colombia. Revista SISTEMAS de la Asociación Colombiana de Ingenieros de Sistemas ACIS, No.72, Julio-Septiembre de 1997

DECRETO 2269 (16 de Nov. 1993) Organización del Sistema Nacional de Normalización, Certificación y Metrología. Ministerio de Desarrollo Económico, República de Colombia.

DÍAZ R., Iván. "Ingeniería del Software y el Control Total de la Calidad", Ensayos en Administración, Universidad del Rosario. 1995.

DÍAZ, José Miguel. "Resultado del estudio de necesidades de tecnología de la información". ACUC Noticias Edición 158. 1995.

FAIRLEY E., Richard. Ingeniería de Software. Editorial McGraw-Hill, 1988

GARZÓN, Héctor V. Curso: Validación y Verificación del Software, Aseguramiento de la Calidad del software, Universidad Industrial de Santander UIS, 1995

HUMPHREY, Watts S. A Discipline for Software Engineering, SEI - Series in Software Engineering, Addison Wesley Publishing Company, 1995

KAN, Stephen H. Metrics and Models in Software Quality Engineering. Addison-Wesley Publishing Company, 1995

KHOSHGOFTAAR, Taghi M. Using Process History to Predict Software Quality. IEEE COMPUTER, abril 1998, pp 66-74

LAWRENCE, Shari. Evaluating Software Engineering Standards. IEEE COMPUTER, septiembre 1994, pp 71-79

Norma IEEE 730-89. Estándar de Planes de Garantía para el Desarrollo de Software de Calidad

Norma Técnica Colombiana NTC 3585. ICONTEC 1.994: Sistema de Procesamiento de la Información. Auditoría. Programa de Aseguramiento de la Calidad para el Software previamente desarrollado en aplicaciones no críticas.

Norma Técnica Colombiana NTC - ISO 8402 . ICONTEC 1.990: Administración de la calidad y aseguramiento de la calidad. Vocabulario

Norma Técnica Colombiana NTC - ISO 9000-3. ICONTEC 1.994. Normas para la administración de la calidad y aseguramiento de la calidad Parte 3: directrices para la aplicación de ISO 9001 al desarrollo, suministro y mantenimiento del software.

Norma Técnica Internacional ISO IEC 12207 1.995. Tecnología de la Información. Procesos del Ciclo de Vida del Software

PERRY, William E. Quality Assurance for Information Systems: Methods, Tools and Techniques, QED Technical Publishing Group, 1991

PRESSMAN, Roger. Ingeniería del Software un enfoque practico, Tercera edición. McGraw-Hill, 1993.

QUERUBÍN, Rodrigo. Guías generales sobre desarrollo de Software. Primera edición. Panamericana Formas e Impresores S.A. 1996.

SAIEDIAN, Hossein and KUZARA, Richard. SEI Capability Maturity Model's Impact on Contractors. IEEE COMPUTER, January 1995, pp 16-25

SANDERS, Joc. Software Quality, Primera edición. ACM Press, 1995.

STARK, George and DURST, Robert. Using Metrics in Management Decision Making IEEE COMPUTER, septiembre 1994, pp 42-48

STEBBING Lionel, Quality Assurance, Segunda edición. Ellis Horwood Limited, 1991.

WINBLAND I. Ann. Software orientado a objetos. Addison-Wesley. Wilmington USA, 1993.

DIRECCIONES DE INTERNET

SEI. Software Engineering Institute. Adscrito a la Universidad Carnegie Mellon, de Estados Unidos. Sitio oficial del CMM:
<http://www.sei.cmu.edu>

SQI. Software Quality Institute. Ubicado en Australia, es uno de los sitios autorizados para divulgar información sobre el proyecto ISO/SPICE:
<http://www.sqi.gu.edu.au/spice>

**Correos
de Colombia**



Adpostal

**CAMBIAMOS PARA SERVIRLE MEJOR
A COLOMBIA Y AL MUNDO**

Estos son nuestros servicios utilícelos!

VENTA DE PRODUCTOS POR CORREO
SERVICIO DE CORREO NORMAL
CORREO INTERNACIONAL
CORREO PROMOCIONAL
CORREO CERTIFICADO
RESPUESTA PAGADA
POST EXPRESS
ENCOMIENDAS
FILATELIA
CORRA
FAX

LE ATENDEMOS EN LOS TELEFONOS
2438851 - 3410304 - 3415534
980015503
FAX 2833345

**Cuente con nosotros
Hay que creer en los Correos de Colombia**