

# **Capítulo 7**

## **Estándares Y Métricas**

## **7.1 La Calidad en la Creación de Software.**

En este capítulo se hace referencia al proceso de calidad en la creación de software estándares, medidas referentes a la calidad etc.

Dado que el Plan de Acción es un proyecto diseñado para tener un alcance nacional y satisfacer un amplio número de usuarios, es importante tener el conocimiento previo de algunos conceptos, medidas y normas referentes a la calidad, como por ejemplo: conocer qué son los estándares de software y como trabajan, qué se necesita para cumplir con un estándar de calidad etc.

La información de este capítulo esta dirigida al grupo u organismo que tendrá a su cargo el desarrollo de los módulos de este Plan de Acción; Con dicha información se pretende que el software que se diseñe en los diferentes módulos cumpla con requisitos de calidad y satisfaga un estándar.

### **7.1.1 La gestión de la Calidad.**

La gestión de la calidad se puede entender, por una parte, como el conjunto de actividades y medios necesarios para definir e implementar un sistema y, por otra, responsabilizarse de su control, aseguramiento y mejora continua. El control está dirigido al cumplimiento de requisitos, el aseguramiento a inspirar confianza en que se cumplirá el requisito pertinente y al mejoramiento al aumento de su eficiencia y eficacia.

La actual tendencia internacional está fundamentada en la aplicación y certificación sobre la base de las normas ISO 9000 que suponen un lenguaje común, adoptado ya por un elevado número de países.

Además de las normas ISO 9000, para lograr una efectiva gestión de la calidad es necesario compatibilizarlas con otras normas específicas adecuadas al tipo de actividad que desarrollan. Tal es el caso de las empresas de la industria del software donde se usan metodologías tan extendidas como el CMM (Modelo de Madurez de Capacidades por sus siglas en inglés) con la cual el mundo del software ha creado

su propia línea de trabajo en la gestión de la calidad, trabajando sobre los procesos de producción de software como medio de asegurar la calidad del producto final.

La naturaleza intangible de este negocio y el hecho de que el software constituye un producto del conocimiento de difícil estandarización, hace necesaria la aplicación de otros modelos que prevén la inclusión de la gestión del conocimiento y su integración a los modelos de calidad.

### **7.1.2 Estándares y Métricas.**

Para Crear un sistema web que involucre a diferentes personas e instituciones y que interactúe con una gran cantidad de niños y maestros, como es el caso del Plan de Acción propuesto en este documento, es necesaria la definición de estándares a nivel de toda la aplicación.

Según *The American Heritage Dictionary*, el término **estándar** significa: “Un grado o nivel de requerimiento, excelencia o logro”. Como adjetivo significa: “Ampliamente reconocido como un modelo de autoridad o excelencia”.<sup>1</sup>

En el contexto en que se utiliza generalmente el término en Ingeniería, un estándar es una especificación formal de cómo llevar a cabo una determinada actividad, aceptada voluntariamente y de manera consensuada por un universo específico de personas e instituciones relacionadas con el desarrollo de tal actividad, motivados por beneficios comunes.

Como ejemplos conocidos de estándares se tiene: el uso de corriente alterna de 110 v en aparatos electrodomésticos; el uso del sistema métrico decimal para expresar medidas de longitud, superficie, volumen, peso, masa, etc.; las reglas ortográficas y sintácticas de un idioma; la calificación de exámenes académicos con notas de 1.0 a 10.0, las convenciones de la programación orientada a objetos entre otros.

---

<sup>1</sup> <http://www.houghtonmifflinbooks.com/ahd/100words.shtml>, *American Heritage Dictionary*, 2005

Para la administración de estándares existen instituciones corporativas, nacionales e internacionales, encargadas de la formulación y el mantenimiento de estándares.

Entre estas instituciones se tiene:

**NBS            National Bureau of Standards.**

Es una división del Ministerio de Comercio de los Estados Unidos que tiene bajo su responsabilidad directa el desarrollo de estándares para uso del gobierno federal. Todos los suministros al gobierno se sujetan a estos estándares.

**CONACYT    Consejo Nacional de Ciencia y Tecnología.**

Es una institución salvadoreña autónoma que tiene dentro de sus funciones el definir estándares técnicos y más específicamente de calidad, como las ISO-9000, para la comunidad empresarial salvadoreña. Es el representante de El Salvador ante ISO.

**ANSI           American National Standards Institute.**

Es una reconocida institución estadounidense que se encarga de coordinar las propuestas y discusiones sobre nuevos estándares; y de coordinar los estándares vigentes y voluntariamente aceptados en los Estados Unidos.

Es una federación de más de 150 gremios empresariales y asociaciones profesionales; junto con más de 1000 empresas que le pagan membresías anuales para apoyar sus actividades. Provee las facilidades necesarias para el desarrollo y aprobación de estándares que gozan de un consenso nacional.

Desde su nacimiento, ANSI ha publicado mas de 3,000 estándares nacionales para los Estados Unidos, decenas de los cuales están directamente relacionados con la industria de las computadoras. ANSI es el representante de los Estados Unidos ante ISO.

**ISO                    International Standards Organization.**

Es una institución semejante a la ANSI; pero aceptada a nivel internacional.

La necesidad y los beneficios de formalizar y documentar los procesos administrativos de una empresa, es algo que está relativamente claro para todo el mundo desde hace mucho tiempo. Sin embargo, no es tan común en las prácticas de desarrollo de software.

**7.1.3 Utilidad de los estándares en proyectos informáticos.**

Al implementar el Plan de Acción, será necesaria la documentación de los procesos informáticos para asegurarse que éstos sean entendidos por los otros elementos que interactúan con el sistema en su desarrollo, mantenimiento y ejecución.

Algunos beneficios para el administrador del proyecto, al hacer uso de estos estándares, son los siguientes:

- Mejoran la comunicación del equipo de trabajo, independientemente de la ubicación y labor específica de cada uno de ellos.
- Establecen formalmente índices para medición del proyecto.
- Proveen las bases para una gestión administrativa y técnica del proyecto.

Al equipo de trabajo que se encuentre desarrollando el proyecto en cada una de sus fases, le es de gran utilidad percibir estándares y métodos de gestión administrativa interna que sean: claros, consistentes y documentados. Algunos beneficios son:

- Claridad. Saber específicamente hacia donde se dirigen bajo una dirección clara y consistente.

- **Productividad** al tener la facilidad de desempeñar sus actividades dentro de un marco aceptado, lo que normalmente le permitirá tener mayor efectividad, eficiencia y productividad.

#### **7.1.4 Tipos de estándares a considerar.**

**Estándares internos.** Cómo el equipo de trabajo tiene que operar, tiempos de ejecución, metodología y lenguajes de programación entre otros.

**Estándares de Hardware.** Estructuras de medios magnéticos, de medios ópticos, nomenclatura de teclados, tipos de arquitectura de hardware a considerar.

**Estándares de Intercambio de información.** Estos son destinados a facilitar el intercambio de información electrónicamente registrada, entre puntos de captura de información e instalaciones de procesamiento de datos. Cubren aspectos como estructuras de archivos; estructuras de codificación, protocolos de transmisión de datos etc.

#### **Estándares de Métodos.**

Se trata de la documentación, de cómo hay que hacer las actividades propias del sistema o en este caso del Plan de Acción. La estandarización de métodos tiene diversas utilidades que deben ser del conocimiento del administrador del proyecto.

- **Control de Calidad.** La calidad de los servicios que provee el sistema, puede ser delimitada y controlada a través de definir pasos específicos con que el programador, el analista y el operador deben seguir para obtenerlos; y la manera en que deben documentarlos.
- **Uniformidad.** De esta forma se pueden intercambiar datos y códigos entre individuos, ya que todos tienen el mismo marco de referencia.

- **Claridad.** Facilita las funciones de supervisión, el establecimiento de objetivos de corto plazo y la entrada en acción de nuevos integrantes del equipo de trabajo.
- **Reducción de Costos.** Se reducen los costos de desarrollo, operación y mantenimiento del sistema y de los módulos creados.
- **Intercambio de Información.** Cuando varias instituciones o grupos de usuarios programadores usan los mismos estándares, el intercambio de información se facilita.
- **Comunicaciones.** Se facilitan al usar los mismos métodos y tecnologías en todo el proyecto informático.

### **Estándares de rendimiento.**

Son valores que indican cuál es la producción mediana esperada, normalmente por unidad de tiempo. Por ejemplo cantidad de formularios digitados por hora, módulos de baja complejidad programables por semana.

La razón principal de desarrollar estándares de métodos es permitir el desarrollo de estándares de rendimiento. Es claramente imposible desarrollar un estándar de rendimiento para determinada actividad, si las personas que la ejecutan no lo hacen de manera compatible con la de otros individuos asignados a la misma actividad.

Los estándares de rendimiento son deseables para varios propósitos:

- **Estimación de costos.** Esto es vital para estar correctamente informados del costo de los requerimientos.
- **Programación de actividades.** Esto es esencial para poder establecer duraciones razonables y para presupuestar los recursos requeridos.
- **Control de tiempos y costos.** Permite al administrador del proyecto elegir entre variar los costos o la duración del proyecto para cumplir con sus metas.

- **Asignación de Costos.** Los estándares de rendimiento permiten estimar los costos de las actividades del proyecto y distribuirlos razonablemente entre los usuarios o instituciones que tendrían relación.

## **7.2 Estándares para el análisis y desarrollo del sistema.**

La idea fundamental detrás del énfasis en el establecimiento y mantenimiento de estándares y documentación, es la necesidad de contar con una metodología determinada, entendiendo ésta como el conjunto de operaciones específicas que permiten conocer con acierto la manera y secuencia con la que deberá efectuarse un determinado trabajo.

### **7.2.1 La Normalización.**

La necesidad de normalizar actividades en un proyecto de sistemas o un centro de cómputo presenta las siguientes características y ventajas:

- Es una solución lógica y natural para contrarrestar el caos normal de las primeras etapas.
- Es un prerrequisito ineludible para la descentralización.
- Incrementa la posibilidad de éxito en el proyecto bajo sus tres variables; características del sistema, tiempo y dinero.
- Garantiza que se pueda dar la continuidad debida a los proyectos.
- Asegura el buen funcionamiento de la operación diaria.
- Provee un mejor o más rentable control de consumo de recursos.

Por lo tanto, la normalización de todas las actividades normalizables de un proyecto o unidad informática debe ser objetivo constante a cualquier tarea que se realice. Una actividad es normalizable y debe ser lo normal si:



- Se repite periódicamente de igual o similar forma.
- Es realizada por distintas personas, en diferentes momentos y ubicaciones.
- Exige un procedimiento ordenado y único.
- Tiene la posibilidad de ser retomada por una persona diferente a quien la comenzó.
- Es susceptible a ser diagramada, escrita o enumerada.
- Obedece a una política interna de trabajo.
- Forma parte de una serie de actividades o elementos de igual naturaleza.
- Demanda un estricto control por el consumo de recursos
- En general, se sabe o se presume que debe llevarse a cabo en forma similar o compatible por diversas personas, con diferente preparación y experiencia, en distintas ocasiones y posiblemente en varios lugares.

No obstante, para que la normalización sea funcional, debe cumplir los siguientes requisitos:

- Se debe involucrar en su establecimiento al personal idóneo en cada caso, solicitándole un primer borrador y discutiéndolo con esas personas, y otras relevantes para obtener en la medida de lo posible un consenso.
- No debe ser tan complicada, ni oscura que impida su correcta aplicación por todos.
- Debe mantenerse lo más breve, sencilla y acorde a los recursos e idiosincrasia de la institución donde se usará.
- Debe ser paulatina, con un orden de prioridades establecido por el administrador, de forma que se pueda llevar a la par de las tareas normales.
- Debe ser clara y sin posibilidad de interpretaciones ambiguas.

### **7.2.2 La Documentación.**

La documentación es un caso especial de la normalización o fijación de estándares, por cuanto guarda precisamente el conocimiento, generado interna o externamente

que ayuda a la buena marcha del proyecto informático y garantiza la continuidad del mantenimiento de aplicaciones.

Se debe asignar a un responsable de administrar la documentación, sobre todo de los sistemas generados internamente o por consultores externos, pero es muy importante que cada analista o programador que desarrolle o de mantenimiento a un sistema, elabore o actualice, según sea el caso, la documentación pertinente, siguiendo los estándares establecidos para tal fin.

La documentación de sistemas, si bien debe complementarse con un manual de operación y de usuario, desde el punto de vista del analista exclusivamente, incluye, entre otras cosas, lo siguiente:

- Diagrama funcional de todo el sistema.
- Descripción de procesos, detallando las entradas, salidas y el proceso mismo.
- Diagramas de flujo de los procesos y algoritmos fundamentales.
- Diagramas entidad-relación.
- Esquema de los modelos semánticos asociativos.
- Estructura de datos y características físicas y lógicas de los archivos usados.
- Detalles de las condiciones especiales de ejecución, tales como banderas, palabras clave, prerequisites, uso de recursos específicos.
- Descripción de interfaces con otros sistemas o aplicaciones.
- Bitácora de cambios, dentro de los mismos códigos fuente con responsables, fecha y descripción del cambio.

Por su parte el manual del usuario deberá incluir, al menos:

- Descripción general del sistema.
- Objetivos del sistema.
- Definiciones de los usuarios.
- Convenciones utilizadas y consideraciones generales.
- Conceptos generales a utilizar.

- Guía de operación, que incluye la jerarquía de funciones y la ejecución de menús, de forma escrita y gráfica.
- Mensajes de error y acciones correctivas.
- Glosario de términos técnicos.

## **7.3 Creación de módulos de software.**

### **7.3.1 Herramientas para la creación de módulos.**

#### **Paginas Web**

Página Web: Archivo hecho con el lenguaje de programación HTML. Generalmente, contiene textos y especificaciones que le indican en donde se deben desplegar las imágenes o los archivos multimedia. Cada página Web es un archivo creado en HTML con su propia dirección en Internet (URL). La Web (o world wide web) es hoy en día el principal servicio de Internet. Permite que sus páginas sean vistas por muchas personas, con diferentes tipos de computadoras, con diferentes sistemas operativos y en cualquier parte del mundo.

#### **Servicios Web (Web Services):**

Un servicio Web es una colección de protocolos y estándares que sirve para intercambiar datos entre aplicaciones. Distintas aplicaciones de software desarrolladas en lenguajes de programaciones diferentes y ejecutadas sobre cualquier plataforma pueden utilizar los servicios Web para intercambiar datos en redes de ordenadores como Internet. La interoperabilidad se consigue mediante la adopción de estándares abiertos. Las organizaciones OASIS (Organization for the Advancement of Structured Information Standards) y W3C (World Wide Web Consortium) son los comités responsables de la arquitectura y reglamentación de los servicios Web.

### 7.3.2 Estándares empleados en la creación de módulos.

**XML** (eXtensible Markup Language, lenguaje de marcado ampliable o extensible)

Es un subconjunto simplificado del SGML (Lenguaje de Marcación Generalizado) el cual fue diseñado principalmente para documentos Web. Deja a los diseñadores crear sus propias “etiquetas” o "tags" (Ej: <libro>), habilitando la definición, transmisión, validación, y la interpretación de datos entre aplicaciones y entre organizaciones. Es con toda probabilidad el futuro estándar para el intercambio de información.

**SOAP** (Simple Object Access Protocol, Protocolo de Acceso de Objeto Simple)

Es un protocolo estándar creado por el W3C que define cómo dos objetos en diferentes procesos pueden comunicarse por medio de intercambio de datos XML. SOAP es uno de los protocolos utilizados en los servicios Web.

**WSDL** (Web Services Description Language, Lenguaje de Descripción de Servicios Web)

Es un formato XML que se utiliza para describir servicios Web. WSDL describe la interfaz pública a los servicios Web. Está basado en XML y describe la forma de comunicación, es decir, los requisitos del protocolo y los formatos de los mensajes necesarios para interactuar con los servicios listados en su catálogo. Las operaciones y mensajes que soporta se describen en abstracto y se ligan después al protocolo concreto de red y al formato del mensaje.

### **Porqué utilizar Paginas Web, Servicios Web y XML.**

#### **Paginas Web**

Las ventajas de utilizar Paginas Web para la creación de módulos son las siguientes:

- Pueden desarrollarse en cualquier lenguaje de programación, plataforma o sistema operativo.
- Pueden alojarse en cualquier plataforma o sistema operativo.
- Pueden utilizarse en cualquier plataforma o sistema operativo.

Las páginas Web deben ser generadas dinámicamente según sea la interacción que se da con el usuario. Para generar páginas dinámicamente existen muchos lenguajes de programación que se encarga de ello, por ejemplo asp y jsp.

Dado que en el usuario con lo que interactúa es con páginas Web, es posible que no todos los módulos tengan que ser desarrollados con el mismo lenguaje de programación. Esto es una gran ventaja para la generación de módulos ya que el desarrollador no está obligado a aprender un nuevo lenguaje para crear módulos.

### **Servicios Web.**

Ya que las páginas Web pueden generarse desde cualquier lenguaje de programación, es necesario que la comunicación entre los módulos y el sistema central sea independiente también del lenguaje y sistema operativo. Para esto es necesario utilizar los servicios Web.

### **Archivos XML.**

Para que cada módulo pueda exponer que es lo que realiza, cuantos ejercicios tiene, que niveles posee, el nombre del módulo y mucha más información es necesario que esta sea guardada en alguna parte. Una alternativa es por medio de bases de datos, otra es por texto plano y otra es por medio de archivos XML. La ventaja de utilizar el XML es que es un estándar entonces es posible guardar en cualquier sistema operativo, además que la mayoría de lenguajes de programación puede entenderlos.

### **7.3.3 Estándares usados en el funcionamiento interno.**

Para poder hacer el enlace Sistema Central – Módulo son necesarios tres elementos:

1. Una dirección URL de donde se encuentra alojado el módulo.
2. El nombre del módulo para poder generar el hipervínculo.
3. Una imagen alusiva al módulo para también poder generar un hipervínculo.

Con la dirección URL el Sistema Central podrá acceder al archivo info.xml para generar un menú contextual que presentara el contenido del modulo, ejercicios disponibles, etc.

Este archivo debe estar ubicado en la raíz de la dirección URL del modulo, además deberá seguir el siguiente estándar para que el Sistema Central pueda leerlo sin dificultad:

#### Modulo:

ModuloID: Código de identificación del modulo.

Nombre: Nombre del modulo.

URLImagen: Ubicación de la imagen identificativa del modulo.

Descripción: Amplia descripción de lo que hace el modulo.

Área: Área a la que pertenece el modulo educativo.

Tema: Tema al que pertenece el modulo educativo.

#### Ejercicio

EjercicioID: Código de identificación de los ejercicios que posee el modulo.

Nombre: Nombre del ejercicio

Nivel: El nivel de dificultad.

URL: Ubicación física, dirección URL, en la que se encuentran la página de inicio del ejercicio.

Los servicios Web publicados en el Sistema Central son los siguientes:

#### 1. InfoUsuario:

UsuarioID: Código de identificación del usuario

Nombre: El nombre del usuario,

#### 2. ResultadoUsuario:

UsuarioID: Código de identificación del usuario

ModuloID: Código de identificación del modulo

EjercicioID: Código de identificación del ejercicio realizado

Resultado: Resultado del ejercicio

### **Distribución de carpetas.**

Se recomienda trabajar cada ejercicios de forma modulada es decir que cada carpeta tenga la misma distribución y nombre de carpetas.

- Directorio bdd.  
Donde se guardara información como base de datos
- Directorio img.  
En esta carpeta se guardarían todas las imágenes que necesiten las páginas Web.
- Directorio cmp.  
Componentes que se utilizarían para generar páginas Web dinámicamente, estos componentes pueden ser archivos con extensión .dll para herramientas .net o archivos de extensión .class para tecnología java.
- Archivo index.html.  
Para iniciar todos los ejercicios con una pagina del mismo nombre.

## **7.4 Las Normas ISO 9000 y CMM.**

Dos de las normas de estandarización y certificación de la calidad del software son las ISO 9000 y CMM, las cuales se presentan a continuación con el propósito de conocer las mejores prácticas de las empresas que desarrollan software de alta calidad a nivel internacional y tomar buenas prácticas para el desarrollo del proyecto.

### **7.4.1 ISO 9000.**

ISO 9000 proviene de un organismo internacional creado en Ginebra, Suiza, en 1946 y cuyo nombre en inglés es *International Organization for Standarization*<sup>2</sup> (cuyo

---

<sup>2</sup> <http://www.iso.org>, International Organization for Standardization 2006

acróstico sería IOS), pero sus fundadores prefirieron darle el nombre griego ISO que significa Igual, para facilitar la comprensión del significado que sus creadores quisieron darle.

*La idea es: ofrecer un modelo de aseguramiento de la calidad, capaz de garantizar que la empresa que lo utilice pueda entregar, cuantas veces le ponga en operación, productos de la misma calidad.*

Para lograr la estabilidad de los procesos, ISO desarrolla un conjunto de cinco normas, cada una con un área específica. Las normas 9001, 9002, 9003 o 9004.

- La Norma 9001 es un modelo de aseguramiento de la calidad para compañías que se dedican al diseño, desarrollo, producción, instalación de productos y servicios. Esta es la norma utilizada en el desarrollo de software de calidad.
- La Norma 9002 es un modelo de aseguramiento de la calidad para compañías que se dedican a los mismos procesos de la 9001, pero que no incluyen el diseño y desarrollo de productos.
- La Norma 9003 es aplicable a empresas que se dedican a inspecciones y pruebas finales, como edificios de prueba, ensayos térmicos o simples contratos de agregación de valor.
- La Norma 9004 es una guía para la implementación de un sistema de gerencia de calidad.

Cada uno de estos procedimientos reúne suficientes requisitos para que, adecuadamente usados, permitan la perfecta repetitividad de los procesos de elaboración de los productos y servicios que interesan a sus clientes.

La finalidad de las ISO es asegurar que las empresas tengan un eficiente grado de control en sus procesos productivos y de calidad.



#### **7.4.1.1 Estructura de las normas ISO-9000.**

Para lograr la confiable repetitividad de los procesos, se requiere a lo menos cuatro condiciones básicas para desarrollar un sistema de calidad que involucra la estructura organizativa, responsabilidades, procedimientos y recursos de implementación.

1. Una determinación firme de la gerencia general del proyecto.
2. Un sistema de procedimientos que especifique quien debe hacer que y cuando
3. Instrucciones de trabajo sobre como hay que hacer cada tarea.
4. La forma en que todo lo anterior debe documentarse.

En el ciclo de calidad del sistema propuesto en este trabajo de investigación se han identificado los siguientes aspectos:

- Comunicar los objetivos del proyecto a cada una de las áreas o personas involucradas que vayan en busca de la calidad del sistema.
- Organizar los recursos a utilizar acordes a la política de calidad de cada una de las fases del proyecto.
- Reconocimiento de la necesidad de garantizar la calidad del sistema.
- Asegurar cumplimiento de los objetivos a corto plazo de cada entidad relacionada al proyecto.
- Medir, auditar, evaluar los resultados y tomar acciones correctivas de las no conformidades entre lo descrito y la práctica diaria.

#### **7.4.2 CMM.**

El Modelo de Madurez de Capacidades, es un modelo para la mejora organizacional que describe una serie de normas para planear, desarrollar y administrar el desarrollo y mantenimiento del software. Es una guía desarrollada por un grupo de profesionales donde quedan reflejados conceptos de administración de proyectos y mejoramiento de la calidad del software. No es una descripción de la organización

ideal, ni algo que reemplace a una buena administración o a gente capaz, pero su aplicación a las áreas clave de una organización de tecnología mejora la oportunidad de la organización para cumplir con sus compromisos de costos, calendario, funcionalidad y calidad del producto.<sup>3</sup>

La meta del modelo es tener procesos que sean repetibles, definidos, administrables y optimizados.

Algunos de los beneficios que provee son: un proceso de desarrollo estándar, una mejora en la comunicación, dar consistencia entre las diferentes actividades, reduce rehacer el trabajo y mejora la predicción de la organización en la ejecución de sus proyectos.

Al desarrollar bajo procedimientos estandarizados, se posee la capacidad y la infraestructura de Ingeniería de Software necesarias para desarrollar los proyectos tecnológicos, de manera conjunta, integrando partes de un proyecto de mayor envergadura, o bien realizarlos de manera integral e individualmente; capitalizando experiencias y aportando soluciones rápidas y confiables para cualquier tipo de demanda puntual.

#### **7.4.2.1 Niveles del CMM.**

El CMM está compuesto de 316 prácticas claves agrupadas en 18 áreas y distribuidas en una jerarquía de cinco niveles, a través de los cuales una organización progresivamente alcanza mayor calidad, productividad y menores costos en el desarrollo de software. Los niveles progresan desde el 1, que representa el estado caótico, hasta el nivel 5, que representa el estado de optimización continua.

---

<sup>3</sup> <http://www.sei.cmu.edu/>. Software Engineering Institute 2006

Estos niveles son:

1. Inicial
2. Repetible
3. Definido
4. Administrado
5. Optimizado

Una organización en nivel 1, en el cual se encuentran la mayoría de los grupos de desarrollo en el mundo, produce software utilizando una aproximación de tanteo y error. Una organización en nivel 5 utiliza las mejores prácticas de ingeniería disponibles en el planeta, hace uso de procesos controlados, medibles y en continuo mejoramiento. Es altamente madura y sistemáticamente está en capacidad de producir software de alta calidad. Según estadísticas del SEI, el tiempo promedio para avanzar entre los niveles de madurez es el siguiente:

- De nivel 1 a nivel 2, 23 meses
- De nivel 2 a nivel 3, 22 meses
- De nivel 3 a nivel 4, 28 meses
- De nivel 4 a nivel 5, 17 meses

Cada uno de los niveles del CMM se orienta a un aspecto específico y explica detalladamente las áreas clave del proceso que se deben considerar. El nivel 2, se enfoca hacia la administración de proyectos; el nivel 3, hacia la definición e integración de un proceso consistente; el nivel 4, en la calidad del producto y el proceso, así como en la realización de mediciones exhaustivas; el nivel 5, se enfoca al mejoramiento continuo del proceso y a la prevención de defectos que pudieran causar riesgos en la operación.

Los cinco niveles pueden ser descritos brevemente como:

**Nivel 1. Inicial.** En este nivel, los procesos y métodos de ingeniería no se encuentran definidos. Por esa razón, los proyectos son adelantados de manera incoherente, incontrolada y poco profesional. Carencia de procedimientos formales,

estimaciones de costo, planes del proyecto, mecanismo de administración para asegurar que los procedimientos se siguen. El éxito es eventual y depende del comportamiento heroico de algunos individuos, cuando estos poseen algún nivel de conocimiento. La mayoría de los grupos de desarrollo de software en el mundo operan a este nivel.

**Nivel 2. Repetible.** Se establecen algunos procesos y métodos de ingeniería a nivel de proyectos, aún rudimentarios. Son establecidas las actividades básicas para la administración de proyectos de software para el seguimiento de costos, programación y funcionalidad. El éxito está en repetir prácticas que hicieron posible el éxito de proyectos anteriores. Por lo tanto hay fortalezas cuando se desarrollan proyectos similares y gran riesgo cuando se enfrentan nuevos desafíos. Involucra las siguientes áreas

- Administración de requerimientos.
- Aseguramiento de calidad del software.
- Sub-contratación de desarrollo de software
- Administración de proyectos.
- Administración de la configuración.
- Seguimiento y control de proyectos.

**Nivel 3. Definido.** Los procesos, actividades y métodos relacionados con la ingeniería y administración de proyectos se encuentran documentados, estandarizados y contruidos alrededor de un marco integrado para toda la compañía. Todos los integrantes de la organización los utilizan en su trabajo diario. Involucra las siguientes áreas

- Enfoque a los procesos de software.
- Definición de los procesos de la organización.
- Programa de entrenamiento.
- Administración integrada del software.
- Ingeniería de software.
- Coordinación intergrupala.
- Revisión entre pares.

**Nivel 4. Administrado.** La compañía opera bajo control estadístico de procesos, tanto en procesos como en productos, las medidas detalladas de las actividades del procedimiento y calidad del producto son registradas. Los resultados de los procesos y la calidad de los productos son predecibles. El proceso de software y el producto entregado son cuantitativamente entendidos y controlados. Involucra las siguientes áreas:

- Administración cuantitativa del proceso
- Administración de la calidad del software

**Nivel 5. Optimizado.** En este nivel, las organizaciones se encuentran en un proceso de mejoramiento continuo. Todos los procesos y técnicas modernas están en pie, lo mismo que la administración cuantitativa. La recolección de datos es automatizada y usada para identificar elementos más débiles del proceso, se hacen rigurosos análisis de causas y prevención de defectos. Las organizaciones se enfocan en el mejoramiento a través de técnicas y procesos de prevención de defectos, cambios en tecnología y cambios en procesos. Involucra las siguientes áreas

- Prevención de defectos.
- Administración de tecnología.
- Administración de cambios al proceso.

#### **7.4.3 CMMI.**

El CMMI que es una evolución de CMM y que integra los distintos modelos de calidad.

- Capability Maturity Model for Software (SW-CMM) v2.0 draft C,
- Electronic Industries Alliance Interim Standard (EIA/IS) 731
- Integrated Product Development Capability Maturity Model (IPD-CMM) v0.98.

##### **7.4.3.1 Niveles del CMMI.**

CMMI propone 5 distintos modelos de madurez de las organizaciones:

1. Inicial - Estado inicial donde el desarrollo se basa en la heroicidad y responsabilidad de los individuos.

- Los procedimientos son inexistentes o localizados a áreas concretas.
  - No existen plantillas definidas a nivel corporativo.
2. Gestionado - Se normalizan las buenas prácticas en el desarrollo de proyectos (en base a la experiencia y al método).
- En este nivel consolidado, las buenas prácticas se mantienen en los momentos de estrés.
  - Están definidos los productos a realizar.
  - Se definen hitos para la revisión de los productos.
3. Definido - La organización entera participa en el proceso eficiente de proyecto software.
- Se conoce de antemano los procesos de construcción de software.
  - Existen métodos y plantillas bien definidas y documentados.
  - Los procesos no solo afectan a los equipos de desarrollo sino a toda la organización relacionada.
  - Los proyectos se pueden definir cualitativamente.
4. Cuantitativamente Gestionado
- Se puede seguir con indicadores numéricos (estadísticos) la evolución de los proyectos.
  - Las estadísticas son almacenadas para aprovechar su aportación en siguientes proyectos.
  - Los proyectos se pueden pedir cuantitativamente.
5. Optimizado
- En base a criterios cuantitativos se pueden determinar las desviaciones más comunes y optimizar procesos.
  - En los siguientes proyectos se produce una reducción de costes gracias a la anticipación de problemas y la continua revisión de procesos conflictivos.

#### **7.4.4 Combinación de ISO 9000 y CMM para el Desarrollo de Software.**

Para acometer la gestión de la calidad con resultados intermedios que permitan a la empresa ir obteniendo madurez en la medida en que avanza la implantación de las normas, se propone definir 3 niveles, ya no específicos al proceso de desarrollo del producto como da cuenta el CMM, sino referentes a la gestión de la calidad a nivel de toda la organización:

1. Nivel inicial
2. Nivel de proyecto.
3. Nivel de gestión total de calidad

##### **Nivel inicial.**

Toda empresa de software aplica al menos un mínimo control de la calidad de sus productos y establece algunos parámetros mínimos como requisitos del producto.

Las empresas que trabajan a este nivel se caracterizan por:

- Definición de políticas elementales de contratación.
- Obtención de las especificaciones de los clientes.
- Formación empírica de equipos de trabajo de acuerdo a la disponibilidad de personal que se tenga.
- Compromisos de plazos de entrega sin criterios bien fundamentados.
- Uso de herramientas de ingeniería de software a voluntad de los desarrolladores.
- Empleo de los lenguajes de programación en que se tiene experiencia, al no ser que el cliente exija lo contrario.
- Mínima documentación.
- Acta de aceptación del cliente quien generalmente no está en condiciones de evaluar adecuadamente el producto.
- Deficiente organización del mantenimiento posventa del producto.

### **Nivel de proyecto.**

En este nivel se incluyen: evaluaciones, revisiones y auditorias, normas, procedimientos de control y seguimiento de errores, así como el registro de la información como vía de retroalimentación para la calidad del proyecto. Algunas acciones importantes que no pueden dejar de realizarse son: asegurar que los procesos de ingeniería de software sean usados durante todo el ciclo de vida del proyecto; evaluar mediante métricas, los requerimientos de calidad de los productos; chequear mediante métricas las condiciones anormales en los procesos y eliminarlas antes de terminar el producto; analizar las pérdidas de calidad para definir acciones para el mejoramiento continuo del proceso de desarrollo de software.

A este nivel pudiera organizarse la actividad de la forma siguiente:

1. Establecer políticas y procedimientos generales para: negociación del proyecto, decisión de plazos de entrega, formación del equipo de proyecto, decisión de plataformas de trabajo y lenguajes de programación, documentación mínima necesaria, herramientas comunes e imagen corporativa implícita en el software a desarrollar.
2. Planificar y organizar las tareas a ejecutar, definiendo las métricas del proyecto, los hitos, las pruebas a realizar en cada uno.
3. Ejecución del proyecto.
4. Control de calidad en cada uno de los hitos definidos y al final. (el control incluye la verificación y corrección).

### **Nivel de gestión total de calidad.**

El control de calidad en cada uno de los hitos debe realizarse por personal especialmente dedicado a esta función. Debe escogerse los técnicos de mayor experiencia.

El control de calidad al final, se recomienda lo realicen círculos de calidad creados ad hoc para cada proyecto.

La separación de las actividades de análisis, programación e implantación es de gran ayuda al permitir lograr de forma natural una contrapartida permanente que sirva de



retroalimentación durante todo el proceso de desarrollo. Sin mencionar otros beneficios como el incremento de los niveles de productividad.

Una vez superada la etapa de lograr la calidad en los proyectos, la empresa estará en condiciones de revisar las normas ISO 9000 de manera detallada y comenzar a adecuar sus actividades y procesos a los requisitos mínimos que ellas establecen.

### **7.5 Planificación del Proyecto.**

El propósito de la planificación es establecer un plan razonable para el desarrollo y administración de cualquier proyecto. Los planes razonables están basados en estimaciones realistas del trabajo que permitan establecer los compromisos necesarios para desarrollar el proyecto. La planificación incluye pasos para estimar la cantidad de trabajo y los recursos necesarios.

#### **Supervisión y seguimiento.**

El propósito del seguimiento y supervisión del proyecto es establecer una visibilidad adecuada del progreso real de este, para que la gestión pueda tomar acciones efectivas cuando el proyecto se desvía significativamente de la planificación realizada.

#### **Aseguramiento de calidad.**

El propósito del aseguramiento de la calidad de un proyecto involucra una revisión y análisis del producto de software y las actividades, para verificar que cumplen con los procedimientos y estándares aplicados.

#### **Administración de la configuración.**

El propósito de la administración de la configuración es establecer y mantener la integridad de los productos de software, a través de su ciclo de vida. Esto significa controlar los cambios de los productos, registrar y mantener las bibliotecas de programas.