

Temario:

Unidad V: Controles para la gestión de proyectos de software.

5.1 .- Control para la calidad del software.

5.1.1 .- Conceptos de calidad.

5.1.2 .- La tendencia de la calidad.

5.1.3 .- Aseguramiento de la calidad del software.

5.1.4 .- Revisiones del software.

5.1.5 .- Revisiones técnicas formales.

5.1.6 .- Confiabilidad del software.

5.2 .- Control del avance.

5.2.1 .- La revisión y su importancia.

5.2.2 .- Objetivos de la revisión.

5.2.3 .- El procedimiento de la revisión.

Unidad V: Controles para la gestión de proyectos de software.

5.1 .- Control de calidad del software.

5.1.1 .- Concepto de calidad.

Hoy día, el concepto de Calidad puede considerarse plenamente incorporado al ámbito empresarial. Sin embargo, se observa cierta confusión en empresas y entidades de todo tipo a la hora de manejar los conceptos de Aseguramiento de la Calidad y Calidad Total. La **Calidad** se puede definir como la capacidad de lograr objetivos de operación buscados.

- La norma ISO 8402-94 define la calidad como:
El conjunto de características de una entidad que le otorgan la capacidad de satisfacer necesidades expresas e implícitas.
- La norma ISO 9000:2000 la define como:
La capacidad de un conjunto de características intrínsecas para satisfacer requisitos.

5.1.1 .- Concepto de calidad.

Enfoque Integrador:

El Enfoque Integrador de la Calidad presenta tres perspectivas: La del punto de vista de la empresa o Interna; la del punto de vista del mercado o Externa; y la llamada Global o de Excelencia.

a) **Perspectiva Interna:** Pone énfasis en la eficiencia. Parte del supuesto de que la empresa ofrece productos que le interesan al mercado, por lo que lo importante es elaborar el producto o servicio con especial atención a los costos y la productividad, respetando lo pactado con el cliente de forma tácita o explícita.

b) **Perspectiva Externa:** Pone la eficacia ante la eficiencia; énfasis en los deseos y satisfacción del cliente. En mercados con alto grado de rivalidad entre competidores, fuerte cambio tecnológico y cambios en las preferencias de los consumidores, es necesario centrarse en el cliente, que es quien indicará qué clase de productos necesita, con qué prestaciones y a qué precio.

5.1.1 .- Concepto de calidad.

c) **Perspectiva Global:** Parte de la base de la empresa excelente es aquella que satisface las necesidades de todos los grupos de influencia relacionados con ella y lo hace con criterios de eficiencia. La excelencia pasa de ser un estado a alcanzar a *"una filosofía de trabajo que da lugar a un proceso dinámico de mejora en el que el objetivo es alcanzar la eficiencia y la eficacia"*, cumpliendo al mismo tiempo con las exigencias de los diversos grupos de personas relacionadas con la organización, que son quienes justifican y posibilitan su existencia.

En resumen, este enfoque reúne el concepto de calidad desde el punto de vista del cumplimiento de los estándares impuestos por la empresa que produce en bien o producto, desde el punto de vista de los estándares demandados por el mercado (clientes), y el punto de vista de la excelencia: "hacer las cosas bien a la primera".

5.1.2 .- La tendencia de la calidad.

La Calidad Total es el estadio más evolucionado dentro de las sucesivas transformaciones que ha sufrido el término Calidad a lo largo del tiempo. En un primer momento se habla de Control de Calidad, primera etapa en la gestión de la Calidad que se basa en técnicas de inspección aplicadas a Producción. Posteriormente nace el Aseguramiento de la Calidad, fase que persigue garantizar un nivel continuo de la calidad del producto o servicio proporcionado. Finalmente se llega a lo que hoy en día se conoce como Calidad Total, un sistema de gestión empresarial íntimamente relacionado con el concepto de Mejora Continua y que incluye las dos fases anteriores. Los principios fundamentales de este sistema de gestión son los siguientes:

- Consecución de la plena satisfacción de las necesidades y expectativas del cliente (interno y externo).
- Desarrollo de un proceso de mejora continua en todas las actividades y procesos llevados a cabo en la empresa (implantar la mejora continua tiene un principio pero no un fin).

5.1.2 .- La tendencia de la calidad.

- Total compromiso de la Dirección y un liderazgo activo de todo el equipo directivo.
- Participación de todos los miembros de la organización y fomento del trabajo en equipo hacia una Gestión de Calidad Total.
- Involucración del proveedor en el sistema de Calidad Total de la empresa, dado el fundamental papel de éste en la consecución de la Calidad en la empresa.
- Identificación y Gestión de los Procesos Clave de la organización, superando las barreras departamentales y estructurales que esconden dichos procesos.
- Toma de decisiones de gestión basada en datos y hechos objetivos sobre gestión basada en la intuición. Dominio del manejo de la información.

5.1.2 .- La tendencia de la calidad.

La filosofía de la Calidad Total proporciona una concepción global que fomenta la Mejora Continua en la organización y la involucración de todos sus miembros, centrándose en la satisfacción tanto del cliente interno como del externo. Podemos definir esta filosofía del siguiente modo: Gestión (el cuerpo directivo está totalmente comprometido) de la Calidad (los requerimientos del cliente son comprendidos y asumidos exactamente) Total (todo miembro de la organización está involucrado, incluso el cliente y el proveedor, cuando esto sea posible).

5.1.3 .- Aseguramiento de la calidad del software.

El aseguramiento de calidad del software es el conjunto de actividades planificadas y sistemáticas necesarias para aportar la confianza adecuada en que el producto (software) satisficará los requisitos dados de calidad. El Aseguramiento pretende dar confianza en que el producto tiene calidad.

Aseguramiento de calidad se enfoca en identificar y evaluar los defectos que puedan afectar al software. Si los errores se pueden identificar de forma temprana en el proceso de software, las características del diseño de software se pueden especificar de modo que eliminarán o controlarán los peligros potenciales, al corregir los errores mucho antes en cada etapa es decir durante el proceso, ahorrando esfuerzos, tiempo y recursos.

Sridharan (Sridharan, 2000) indica que mientras el software que se está desarrollando reúne los requerimientos y su desempeño sea el esperado, es preciso que se supervisen las actividades de desarrollo del software y su rendimiento, en distintas oportunidades durante cada fase del ciclo de vida. Este es el papel del aseguramiento de la calidad del software.

5.1.3 .- Aseguramiento de la calidad del software.

Hay tres aspectos muy importantes con relación al aseguramiento de la calidad del software: (Wieggers, 1990)

- La calidad no se puede probar, se construye.
- El aseguramiento de la calidad del software no es una tarea que se realiza en una fase particular del ciclo de vida de desarrollo.
- Las actividades asociadas con el aseguramiento de la calidad del software deben ser realizadas por personas que no estén directamente involucradas en el esfuerzo de desarrollo.

5.1.3 .- Aseguramiento de la calidad del software.

El aseguramiento de la calidad de software comprende una gran variedad de tareas:

- Participar en descripción del proyecto de software.
- Auditar el producto de software para verificar el cumplimiento del proceso de software definido.
- Asegurar que las divergencias en el trabajo de software sean documentadas de acuerdo a los estándares definidos.
- Almacenar cualquier inconformidad y reportarla a la gerencia media.
- Revisiones del software que se aplican durante cada paso del desarrollo del mismo. Las revisiones del software se aplican en varios momentos del desarrollo, tanto en el análisis como en el diseño y la codificación, de manera que puedan ser eliminados cuanto antes.

5.1.3 .- Aseguramiento de la calidad del software.

Para lograr y asegurar la calidad al usuario o cliente, es necesario controlar la calidad en cada una de las etapas de desarrollo, desde el diseño del producto o sistema hasta el mantenimiento. Estas etapas se constituyen como el ciclo de la calidad, habiendo interdependencia y retroalimentación en cada una de ellas. Para satisfacer las necesidades o requerimientos del usuario o consumidor, debe existir calidad en:

- El análisis y diseño del producto informático.
- Abastecimientos de herramientas propias para desarrollo del sistema. Producción.
- Entrega del sistema terminado.
- Venta o servicio.
- Administración general.

5.1.3 .- Aseguramiento de la calidad del software.

El grupo de aseguramiento de calidad participa en la revisión de los productos seleccionados para determinar si se conforman o no a los procedimientos, normas o criterios especificados, siendo totalmente independiente del equipo de desarrollo.

Las actividades a realizar por el grupo de aseguramiento de calidad vienen gobernadas por el plan. Sus funciones están dirigidas a:

- Identificar las posibles desviaciones en los estándares aplicados, así como en los requisitos y procedimientos especificados.
- Comprobar que se han llevado a cabo las medidas preventivas o correctivas necesarias.
- Las revisiones son una de las actividades más importantes del aseguramiento de la calidad, debido a que permiten eliminar defectos lo más pronto posible, cuando son menos costosos de corregir.

5.1.3 .- Aseguramiento de la calidad del software.

También es importante señalar que la calidad de un producto software no se puede referir únicamente a obtener un producto sin errores. La especificación de la calidad del software debe ser más detallada y exacta, y el camino para ello es la formalización de la calidad mediante un modelo de calidad.

Un modelo de calidad es un conjunto de buenas prácticas para el desarrollo del software, enfocado en los procesos de gestión y desarrollo de proyectos, cuyo objetivo es el desarrollo de productos de calidad de manera consistente y predecible.

5.1.4 .- Revisiones del software.

Las inspecciones de software surgen a partir de la necesidad de producir software de alta calidad.

Algunos grupos de desarrollo creen que la calidad del software es algo en lo que deben preocuparse una vez que se ha generado el código. ¡ Error ¡ La garantía de la calidad del software es una actividad de protección que se aplica a lo largo de todo el proceso de ingeniería de software. La SQA (Software Quality Assurance) engloba:

- Un enfoque de gestión de calidad
- Tecnología de Ingeniería de Software efectiva (métodos y herramientas)
- Revisiones técnicas formales que se aplican durante el proceso del software
- Una estrategia de prueba multiescalada
- Un control de la documentación del software y de los cambios realizados
- Un procedimiento que asegure un ajuste a los estándares de desarrollo de software
- Mecanismos de medición y de generación de informes

5.1.4 .- Revisiones del software.

Una revisión es una forma de aprovechar la diversidad de un grupo de personas para:

- Señalar la necesidad de mejoras en el producto de una sola persona o de un equipo
- Confirmar las partes del producto en las que no es necesaria o no es deseable una mejora.
- Conseguir un trabajo de mayor calidad maximizando los criterios de Correctitud y Completitud principalmente .

Existen muchos tipos diferentes de revisiones que se pueden llevar adelante como parte de la ingeniería del software. Cada una tiene su lugar. Una reunión informal durante el almuerzo o en un café es una forma de revisión, si se discuten problemas técnicos. Una presentación formal de un diseño de software a una audiencia de clientes, ejecutivos y personal técnico es una forma de revisión.

5.1.4 .- Revisiones del software.

El proceso de inspección.

Podemos ver a las inspecciones de software como un repaso detallado y formal del trabajo en proceso. Pequeños grupos de trabajadores (4 o 5) estudian el ""producto de trabajo"" independientemente y luego se reúnen para examinar el trabajo en detalle. El ""producto de trabajo"" representa 200 a 250 líneas de código, Requerimientos, diseño y otros productos de trabajo son inspeccionados en un tamaño similar. Los productos de trabajo son considerados en progreso hasta que la inspección y las correcciones necesarias estén completas.

La inspección consiste en seis pasos [Fagan 1986] :

1.- Planificación: Cuando el desarrollador completa su un ""producto de trabajo"", se forma un grupo de inspección y se designa un moderador. El moderador asegura que el ""producto de trabajo"" satisfaga el criterio de inspección. Se le asignan diferentes roles a las personas que integran el grupo de inspección, así como la planificación de tiempos y recursos necesarios .

5.1.4 .- Revisiones del software.

2.- Overview: Si los inspectores no están familiarizados con el desarrollo de l proyecto, un vista general es necesaria en éste momento. Este es un paso opcional, pero no menos importante ya que en esta etapa se dará al grupo de inspección un "background" y el contexto a cubrir por las inspecciones.

3.- Preparación: Los inspectores se preparan individualmente para la evaluación en la reunión, estudiando los productos de trabajo y el material relacionado. Aquí es aconsejable la utilización de listas de chequeos para ayudar a encontrar defectos comunes, y . El tiempo que pueda llevar esta etapa va a depender de cuan familiarizado esté el inspector con el trabajo que debe analizar.

5.1.4 .- Revisiones del software.

4.- Examen: En esta etapa, los inspectores se reúnen para analizar su trabajo individual en forma conjunta. El moderador deberá asegurarse que todos los inspectores están suficientemente preparados. La persona designada como lector presenta el "producto de trabajo", interpretando o parafraseando el texto, mientras que cada participante observa en busca de defectos. Es recomendable que este examen no dure más de 2 horas ya que la atención en busca de defectos va disminuyendo con el tiempo. Al terminar con la reunión, el grupo determina si el producto es aceptado o debe ser retrabajado para una posterior inspección.

5.- Retrabajo: El autor corrige todos los defectos encontrados por los inspectores.

6.-Seguimiento: El moderador chequea las correcciones del autor. Si el moderador está satisfecho, la inspección está formalmente completa, y el "producto de trabajo" es puesto bajo el control de configuración.

5.1.4 .- Revisiones del software.

A partir de estos seis pasos surge la necesidad de la conformación de: objetivos, participantes de la inspección y con que roles, productos de trabajo a inspeccionar y cual deberá ser el resultado de las reuniones de inspección.

- **Objetivos:** El principal objetivo es encontrar defectos en el "producto de trabajo", de esta manera debemos definir cuales serán las metas a alcanzar para el cumplimiento de éste objetivo. En nuestra opinión el establecimiento de éstas metas están en relación directa con el tipo de proyecto, metodologías y herramientas utilizados
- **Grupos de inspección:** Se recomienda formar grupos entre 3 y 6 personas [IEEE STD 1028-1988]. Dentro de éste grupo debe incluirse al autor de los productos de trabajo.
- **Roles:** Además del autor se deberá tener en cuenta al moderador quien dirige la inspección y deberá contar con mayor experiencia que el resto, el lector quien presenta los productos de trabajo en las reuniones y el escriba quien documenta la descripción y ubicación de los defectos encontrados.

5.1.4 .- Revisiones del software.

- Productos de trabajo: El proceso de inspección puede ser aplicado a diferentes tipos de productos de trabajo que pueden encontrarse en un desarrollo de software incluyendo requerimientos, diseño, código, test, guías de usuario y otro tipo de documentación. El estándar de la IEEE no especifica un tamaño , pero los productos de trabajo tienen un tamaño de 10 a 20 hojas para especificación de requerimientos, 200 o 250 líneas de código. En nuestra opinión también se debe tener en cuenta la división natural que pueda tener cada uno de éstos documentos, ya que toda especificación podremos dividirla en partes así como el diseño o el código.
- Resultado de las reuniones de inspección: Los dos resultados principales debe ser: Una lista de defectos a corregir , y un reporte de inspección que describa que es lo que se inspeccionó, quien inspeccionó qué y el número de defectos encontrados.

5.1.5 .- Revisiones técnicas formales.

Objetivos: descubrir errores en la función, lógica o implementación de cualquier representación del software. Verificar el cumplimiento de los requisitos Garantizar el cumplimiento de los estándares. Conseguir un desarrollo uniforme del software Obtener proyectos que hagan más sencillo los trabajos técnicos (análisis que permitan buenos diseños, diseños que permitan implementaciones sencillas, estrategias de pruebas que faciliten éstas,...)

RTFs: son un filtro que permite “purificar” las actividades de ingeniería de software. se aplican en diversos momentos del desarrollo para detectar defectos. Diseño: entre el 50 y el 60% de los errores del desarrollo. Aprovecha la diversidad de un grupo de personas para: señalar la necesidad de mejoras en el producto de ingeniería (diagramas del análisis, diccionario de datos, diseño, código, estrategia de pruebas,...) Confirmar las partes en las que no es necesaria una mejora. Conseguir un trabajo técnico de calidad más uniforme. Efectividad: se calcula que son efectivas en un 75%.

5.1.5 .- Revisiones técnicas formales.

Técnicas

“Estáticas: análisis y chequeo de documentos de requisitos, diagramas de diseño, código fuente, etc.

“dinámicas: pruebas sobre implementación real (sólo pueden Usarse cuando ya se tiene código ejecutable).

Revisiones Técnicas Formales

Se eliminan errores en forma relativamente temprana (barato y fácil de corregir) Cada revisión se conduce en forma de una reunión cuidadosamente planeada y controlada

5.1.5 .- Revisiones técnicas formales.

La reunión de revisión

Entre 3 y 5 personas (grupo pequeño) Preparación previa (2 horas por persona) Especificación precisa (formal o informal)

Estándares de codificación

Duración máxima 2 horas (lento pero cansado) Foco en un segmento específico Participan los revisores y el productor

Directrices para la revisión

Revisar el producto y no al productor Indicar los errores con tino, tono constructivo Mantenerse estrictamente dentro de la agenda No irse por las ramas Limitar el debate Algunos asuntos pueden dejarse para discusión posterior Enunciar problemas no resolverlos Problema debería ser resuelto por el productor Tomar notas (pizarra deseable)

5.1.6 .- Confiabilidad del software.

La IEEE define a la confiabilidad como la habilidad que tiene un sistema o componente de realizar sus funciones requeridas bajo condiciones especificas en periodos de tiempo determinados. La probabilidad o la capacidad de que un sistema de funciones trabajen sin falla en un periodo de tiempo y bajo condiciones o un medio ambiente tambien especifico.

Los principales aspectos asociados con la confiabilidad del software fueron:

- Fiabilidad.
- Disponibilidad.
- Mantenimiento.
- Seguridad.

5.1.6 .- Confiabilidad del software.

Disponibilidad: se refiere a la condición de trabajo que un sistema debe de tener. Si el sistema que se está utilizando llega a ser apagado en algún tiempo anterior al que se tenía planeado por mantenimiento, luego tendrá que ser encendido sin que afecte su rendimiento o se tenga algún problema.

Fiabilidad: en la ingeniería se usa generalmente para asegurar aquella condición de trabajo que permite al usuario realizar sus tareas para que el sistema no llegue a corromperse.

Seguridad: este concepto no solo describe el comportamiento del sistema, también nos define la habilidad que tiene para poder resistir los ataques externos.

Protección: este concepto se refiere a la capacidad del sistema de permitir las fallas de manera inmediata, en caso de que el sistema llegara a fallar existirá alguna manera de proteger la información o las acciones que el sistema realice.

5.1.6 .- Confiabilidad del software.

La confiabilidad del software se encuentra en un etapa de formación de desarrollo y es la característica de rendimiento más costosa de conseguir y difícil de conseguir y de difícil de garantizar. La naturaleza del proyecto ayuda para la formulación de estimaciones de costo y el esfuerzo que asegure la confiabilidad requerida.

Los modelos de confiabilidad del software se usan para caracterizar y predecir el comportamiento importante para directores e ingenieros.

La generación de fallos depende del código desarrollado, tales como tamaño y las características del proceso de desarrollado tales como las tecnologías y herramientas de ingeniería de software usadas.

5.1.6 .- Confiabilidad del software.

La eliminación de fallos depende del tiempo y del perfil operativo. Los modelos de confiabilidad del software son generalmente procesos aleatorios. Estos modelos se pueden dividir en 2 grandes categorías:

1- modelos que predicen la confiabilidad como una función cronológica del tiempo

2- modelos que predicen la confiabilidad como una función del tiempo de procesamiento transcurrido

5.1.6 .- Confiabilidad del software.

La confiabilidad de un sistema es una medida de conformidad dependiendo de la especificación que se encuentra autorizada a su comportamiento, a su vez esta es usada para la prevención o tolerancia de fallos de un sistema y como la confiabilidad en la ingeniería es usada generalmente para asegurar cierto grado de funcionamiento con éxito de un dispositivo o sistema en un ambiente determinado, durante un periodo específico de tiempo. Se puede distinguir entre la confiabilidad de la misión de un dispositivo, el cual es construido para llevar a cabo el funcionamiento de una misión, así como también la operación de confiabilidad de un sistema el cual puede ser o no ejecutado indeterminablemente para el funcionamiento de un cierto grado de misiones.

5.2 .- Control del avance.

5.2.1 .- La revisión y su importancia.

La garantía de calidad o aseguramiento de la calidad consiste en la auditoria y las funciones de información de la gestión. El objetivo de la garantía de la calidad es proporcionar la gestión para informar de los datos necesarios sobre la calidad del producto, por lo que se va adquiriendo una visión más profunda y segura de que la calidad del producto está cumpliendo sus objetivos. Es de esperar, que si los datos proporcionados mediante la garantía de la calidad identifican problemas, la gestión afronte los problemas y aplique los recursos necesarios para resolverlos.

La garantía de calidad del software comprende una gran variedad de tareas, asociadas con dos constitutivos diferentes: los ingenieros de software, que realizan trabajo técnico, y un grupo SQA, que tiene la responsabilidad de la planificación de garantía de calidad.

5.2.1 .- La revisión y su importancia.

En éste marco podemos ver a las inspecciones como una implementación de las revisiones formales del software las cuales representan un filtro para el proceso de ingeniería de software, éstas se aplican en varios momentos del desarrollo y sirven para detectar defectos que pueden así ser eliminados. Freeman y Weinberg [Fre90] argumentan de la siguiente forma la necesidad de revisiones:

El trabajo técnico necesita ser revisado por la misma razón que los lápices necesitan gomas: errar es humano. La segunda razón por la que necesitamos revisiones técnicas es que, aunque la gente es buena descubriendo algunos de sus propios errores, algunas clases de errores se le pasan mas fácilmente al que los origina que a otras personas. El proceso de revisión es, por lo tanto la respuesta a la plegaria de Robert Burns:

¡Que gran regalo sería poder vernos como nos ven los demás!

5.2.1 .- La revisión y su importancia.

Una revisión es una forma de aprovechar la diversidad de un grupo de personas para:

- 1.- Señalar la necesidad de mejoras en el producto de una sola persona o de un equipo
- 2.- Confirmar las partes del producto en las que no es necesaria o no es deseable una mejora.
- 3.- Conseguir un trabajo de mayor calidad maximizando los criterios de Correctitud y Completitud principalmente .

5.2.1 .- La revisión y su importancia.

Existen muchos tipos diferentes de revisiones que se pueden llevar adelante como parte de la ingeniería del software. Cada una tiene su lugar. Una reunión informal durante el almuerzo o en un café es una forma de revisión, si se discuten problemas técnicos. Una presentación formal de un diseño de software a una audiencia de clientes, ejecutivos y personal técnico es una forma de revisión. Sin embargo en éste trabajo nos concentraremos en una revisión técnica formal, que llamaremos Inspección de Software

5.2.1 .- La revisión y su importancia.

Una serie de estudios (TRW, Nippon Electric y Mitre Corp., entre otros) indican que las actividades del diseño introducen entre el 50% y 65% de todos los errores (y en último lugar, todos los defectos) durante el proceso de software. Si embargo se ha demostrado que las inspecciones de software son efectivas en un 75% a la hora de detectar errores .

Con la detección y la eliminación de un gran porcentaje de errores, el proceso de inspección reduce substancialmente el costo de los pasos siguientes en las fases de desarrollo y mantenimiento.

Para ilustrar el impacto sobre el costo de la detección anticipada de errores, consideremos una serie de costos relativos que se basan en datos de costos realmente recogidos en grandes proyectos de software [IBM81]. Supongamos que un error descubierto durante el diseño cuesta corregirlo 1,0 unidad monetaria. De acuerdo a este costo , el mismo error descubierto justo antes de que comience la prueba costará 6,5 unidades; durante la prueba 15 unidades; y después de la entrega, entre 60 y 100 unidades.

5.2.1 .- Objetivos de la revisión.

El objetivo primario de las revisiones técnicas formales (inspección) es encontrar errores durante el proceso para evitar que se conviertan en defectos después de la entrega del software. El beneficio obvio de estas inspecciones es el descubrimiento de errores al principio para que no se propaguen al paso siguiente del proceso de desarrollo del software (catarata de errores de Mizuno).

5.2.1 .- El procedimiento de la revisión.

Proceso genérico de IEEE 1028 para las revisiones formales

IEEE Std 1028 define un sistema común de las actividades para las revisiones “formales” (con algunas variaciones, especialmente para la intervención del software). La secuencia de actividades se basa en gran parte en inspección del software el proceso se convirtió originalmente en IBM cerca Michael Fagan.

Los tipos que diferencian de revisión pueden aplicar esta estructura con grados que varían de rigor, pero todas las actividades son obligatorias para la inspección:

0. [Evaluación de la entrada]: El líder de la revisión utiliza una lista de comprobación estándar de los criterios de la entrada para asegurarse de que las condiciones óptimas existen para una revisión acertada.

1. Preparación de la gerencia: La gerencia responsable se asegura de que la revisión esté apropiadamente resourced con el personal, el tiempo, materiales, y herramientas, y sea conducida según políticas, estándares, u otros criterios relevantes.

5.2.1 .- El procedimiento de la revisión.

2. Planear la revisión: El líder de la revisión identifica o confirma los objetivos de la revisión, organiza a equipo de revisores, y se asegura de que equipan al equipo de todos los recursos necesarios para conducir la revisión.

3. Descripción de los procedimientos de la revisión: El líder de la revisión, o alguna otra persona cualificada, asegura (en una reunión en caso de necesidad) que todos los revisores entienden las metas de la revisión, procedimientos de la revisión, los materiales disponibles para ellos, y los procedimientos para conducir la revisión.

4. Preparación [del individuo]: Los revisores se preparan individualmente para la examinación del grupo del trabajo bajo revisión, examinándola cuidadosamente para *anomalías* (defectos potenciales), la naturaleza de que variará con el tipo de revisión y de sus metas.

5. Examinación [del grupo]: Los revisores satisfacen en un rato previsto de reunir los resultados de su actividad de la preparación y de llegar un consenso con respecto al estado del documento (o de la actividad) que es repasada.

5.2.1 .- El procedimiento de la revisión.

6. Reanudación/carta recordativa: El autor del producto del trabajo (o la otra persona asignada) emprende cualesquiera acciones son necesarias reparar defectos o satisfacer de otra manera los requisitos convenidos en la reunión de la examinación. El líder de la revisión verifica que todos los artículos de acción sean cerrados.

7. [Evaluación de la salida]: El líder de la revisión verifica que todas las actividades necesarias para la revisión acertada se hayan logrado, y que todo hace salir apropiado al tipo de revisión se han concluido.

Cybergrafia.

<http://www.worldlingo.com>

<http://www.monografias.com>

<http://www.mitecnologico.com>

<http://sistemas.itlp.edu.mx>

<http://www.qualitrain.com.mx>

<http://www.eumed.net>

<http://www.calidaddelsoftware.com>

<http://es.wikipedia.org>

<http://www.um.es>