

# La norma ISO 14764

Parte de un trabajo de Asignatura realizado por Samira Lamayzi, dirigido por Francisco Ruiz  
Asignatura: Planificación y Gestión de Sistemas de Información, 1998/1999.

## I.Introducción al estándar.

Éste estándar internacional aclara los requerimientos para el Proceso de Mantenimiento del Software. El Mantenimiento del Software es un proceso primario en el ciclo de vida de un producto software tal como se describe en ISO/IEC 12207, "Tecnología de la información - Software, Parte 1: Los procesos del ciclo de vida del software".

El Proceso de Mantenimiento contiene las actividades y tareas del mantenedor.

Éste estándar internacional es parte de la familia de documentos ISO/IEC 12207 y da una pequeña guía.

La única cláusula obligatoria en éste estándar internacional proviene de ISO/IEC 12207.

Ésta cláusula contiene cosas que se deben hacer y cada una de ellas está marcada dentro de una caja en éste documento. El número de cláusula ISO/IEC 12207 se muestra después de la caja.

Ante todo siempre se pretende conseguir es conocer y ante todo cuales son los principales conceptos relacionados con la gestión de proyectos y realizar un estudio completo de todas las materias abarcadas en la gestión de proyectos.

En muchos proyectos, especialmente aquellos que tienen una vida larga, el mantenimiento del software es con seguridad una de las consideraciones más importantes del proyecto. También se pretende presentar los diversos procesos que se desarrollan al gestionar un proyecto y explicar sus interrelaciones.

Debido al coste del producto y a las restricciones en el tiempo, además de no seguir las mejores prácticas de ISO/IEC 12207, el software se entrega a menudo en un estado imperfecto.

Es necesario ser capaces de corregir los fallos que se encuentran durante su manejo.

A menudo es necesario hacer mejoras al software debido a que los requisitos y las demandas y las necesidades del usuario cambian. El mantenimiento del software puede llegar a ser una parte muy importante de los costes del ciclo de vida.

Éste estándar internacional se profundiza en el estudio del mantenimiento del software y sirve de guía para aquellas impresas o lectores para que puedan familiarizarse y profundizarse más en el mantenimiento del software con este estándar.

Se recomienda a aquellos que no lo estén que lean libros de texto o que se

entrenen en el mantenimiento del software antes de aplicar el estándar internacional.

Mantenimiento de Software se puede hacer combinando herramientas software, métodos y técnicas.

Éste estándar internacional no especifica como implementar o realizar las actividades y tareas en el Proceso de Mantenimiento de Software ya que éste es dependiente del contrato y de la organización. Los requerimientos del Mantenimiento de Software no cambian aunque se cambien las herramientas usadas.

Los puntos que vamos a tocar en este estándar son:

La proporción del alcance, la información para el acuerdo, las referencias a las normativas, términos y definiciones, la aplicación de este estándar internacional, la consideración para la implementación del proceso del mantenimiento del software, la estrategia para el mantenimiento del software y los detalles del proceso de mantenimiento del software y al final hablaremos de un anexo que nos proporciona referencias cruzadas y una comparación entre éstas cláusulas y las de ISO/IEC 12207.

El IEC/TC 56 ha contribuido en gran manera a la realización de éste estándar.

## **II. Tecnología de la información y Mantenimiento de Software**

### **1. Alcance**

¿Para que nos sirve éste estándar, cuales son sus importantes usos, que establecimiento nos proporciona, que establecimientos nos da, a donde nos guía y que efectos proporciona sobre el mantenimiento del software?.

Éstas son las posibles dudas o preguntas que podemos formar al respecto en éste estándar, veamos los siguientes puntos que nos dan el alcance del estándar:

A primera vista podremos entender que lo que se pretende aquí es una gestión del alcance de modo que hay que asegurar que el proyecto incluye todos los trabajos requeridos y sólo éstos y ver lo que está incluido y lo que no en el proyecto. Veamos desde las entradas a las salidas (según el PMI, Project Management Institute o Instituto para la Gestión de Proyectos) los procesos de iniciación, planificación, definición, verificación, y control del alcance por pasos:

- **Iniciación;** se toman de entradas: la descripción del producto, plan estratégico, criterios de selección del proyecto y información histórica. Las técnicas y herramientas son los métodos de selección de proyectos y juicio de expertos. Las salidas que se obtiene son diagramas de proyecto y hay que tener asignado o identificado el administrador del proyecto, restricciones y supuestos. **Planificación del alcance;** se toman de entradas: La descripción del producto, el diagrama del proyecto, las restricciones y los supuestos. Las técnicas que usaremos son análisis del producto, el análisis de la relación

coste/beneficio, identificaremos alternativas y nos someteremos al juicio de expertos. Las salidas son una declaración de objetivos, un plan para la gestión del alcance y un pequeño plan de organización del soporte.

- Definición del alcance; las entradas son la declaración de objetivos, restricciones y supuestos, la información histórica (información de otros proyectos parecidos que hayamos hecho en el pasado) y cualquier salida de la etapa de planificación. Para llevar a cabo esta etapa usaremos descomposición de los objetivos y plantillas para la descomposición de trabajos (podemos usar plantillas porque los trabajos son siempre mas o menos los mismos). El resultado de esta actividad es una estructura de trabajos descompuestos (en los que detallamos qué hacer, quien lo hace).
- Verificación del alcance: En este caso inspeccionamos los resultados del trabajo anterior y la documentación producida para obtener una aceptación formal (a ser posibles que sea por escrito).
- Control del cambio del alcance: Esta actividad esta presente durante todo el ciclo de vida del proyecto. Recibe como entradas la descomposición de trabajos, los informes de rendimiento, las peticiones de cambio y el plan de gestión del alcance. Utilizando un sistema para el control de cambios, un conjunto de métricas y una planificación adicional nos devuelve los cambios concretos y acciones correctivas que tendremos que ejecutar(además de lecciones aprendidas para el futuro).

Éste estándar internacional describe en gran detalle la gestión del Proceso de Mantenimiento descrito en ISO/IEC 12207, y además establece definiciones para los distintos tipos de mantenimiento, y proporciona una guía aplicable a la planificación, ejecución y control, mantenimiento, revisión y evaluación y de forma cercana al proceso de mantenimiento.

Se incluye el mantenimiento para múltiples productos software con los mismos recursos de mantenimiento.

Podremos preguntarnos ¿qué es el mantenimiento dentro de esta norma?, La palabra "Mantenimiento" dentro de éste estándar internacional significa mantenimiento de software a no ser que se indique lo contrario.

Éste estándar internacional proporciona un armazón dentro del cual se pueden ejecutar y evaluar planes de mantenimiento de software generales y específicos así como adaptar el alcance y magnitud de los productos software dados.

Éste estándar internacional proporciona un armazón, terminología precisa y procesos para permitir la aplicación consistente de la tecnología (herramientas, técnicas y métodos) para el mantenimiento de software, y además proporciona una guía para el mantenimiento de software para lo cual será más fácil el seguimiento del estándar.

Las bases para el mantenimiento de software y sus actividades provienen de las definiciones ISO/IEC 12207.

Este grupo define las actividades y tareas del mantenimiento software, proporciona los requerimientos para la planificación del mantenimiento.

No contempla la operación del software ni sus funciones operacionales, p.ej. copia, recuperación de errores, administración del sistema, etc. ... que son llevadas a cabo por

quien maneja el software normalmente.

### **1.1.Propósito**

Éste estándar internacional proporciona una guía sobre la gestión de (o como llevar a cabo el proceso de mantenimiento). Eso da lugar a que dicho estándar proporciona una gran ayuda y facilidad de seguimiento para tener claras ideas sobre el proceso de mantenimiento y su aplicación de modo que identifica cómo el Proceso de Mantenimiento se puede realizar durante la adquisición y operación.

### **1.2.Campo de aplicación**

Éste estándar internacional intenta proporcionar una guía para situaciones con dos individuos y se puede aplicar igualmente cuando los dos pertenecen a la misma organización pero intenta también ser usado por un solo individuo como tareas que se autoimpone.

Éste estándar internacional no está dirigido a usuarios de productos software que no están a la venta a menos que estén incorporados en producto para entregar (ISO/IEC 12207), ni está orientado a productos software que son soluciones a corto plazo de hecho la mayoría de las empresas intentan usar un producto a un cierto tiempo pero más bien largo para ello los productos que desean tener o incorporar deben mantenerse a un tiempo lo mas largo posible eso da lugar al ahorro de costes y por éste ultimo el mantenimiento de la empresa, ni está orientado para productos software personalizados por los usuarios ni a productos para el usuario final. Está orientado a la auto-imposición en los desarrolladores de productos software de procesos para el mantenimiento.

Por ejemplo, organizaciones que puedan desear usar éste estándar internacional cuando mantengan macros ó plantillas usadas en la organización para el procesamiento de palabras.

El mantenimiento se aplica a programas de ordenador, código, datos, y documentación. Se intenta que se aplique a productos software creados durante el desarrollo del producto software.

Ésto puede incluir cosas como software de pruebas, bases de datos de prueba, el Entorno de Pruebas del Software (STE) o el Entorno de Ingeniería de Software (SEE).

Éste estándar internacional está pensado para su uso en todos los esfuerzos de mantenimiento, independientemente del ciclo de vida o del enfoque usado en el desarrollo.

### **1.3.Limitaciones**

Éste estándar internacional describe el esqueleto del Proceso de Mantenimiento Software pero no especifica los detalles de como implementar o ejecutar las actividades y tareas incluidas en el proceso.

En éste estándar internacional hay algunas listas. Ninguna de ellas es exhaustiva, están pensadas como ejemplos, los pasos para aplicar éste estándar internacional están en ISO/IEC 12207.

## **2.Cumplimiento de la normativa**

Un proceso se ajustará a la normativa si satisface los requerimientos de ISO/IEC 12207.

## **3.Referencias a normativas**

Los siguientes documentos de normativas contienen citas que se usarán en éste documento:

ISO/IEC 2382-80: Tecnología de la información - Vocabulario; Parte 20: desarrollo de sistemas.

ISO/IEC 5807: Procesamiento de información - Símbolos para la documentación y convenciones para datos, programas, diagramas de flujo, gráficos de redes de programas y gráficos de recursos del sistema.

ISO 8402: Gestión de la calidad y aseguramiento de la calidad - Vocabulario.

ISO/IEC 9126: Tecnología de la información - Evaluación del producto software - Características de la calidad y guías para su uso.

ISO/IEC 12207: Tecnología de la información - Procesos de ciclo de vida software.

## **4.Definiciones y términos**

Para los propósitos de éste estándar internacional aplicaremos las definiciones y términos dados en ISO/IEC 12207, ISO 8402, ISO/IEC 2382-1 e ISO/IEC 2382-20 además de los siguientes:

### **4.1.Mantenimiento adaptativo**

Se define como la modificación de un producto software hecha después de la entrega, para así mantener el uso de un producto software en un entorno cambiado o cambiante de modo que el mantenimiento adaptativo proporciona mejoras necesarias para acomodarse a los cambios en el entorno en que se ejecuta un producto software.

Éstos cambios son aquellos que deben hacerse para seguir funcionando en el entorno cambiante. Por ejemplo, puede que actualicemos el sistema operativo y que haya que adaptar el software.

#### **4.2.línea base**

Una versión aprobada formalmente de un elemento de configuración, independientemente del medio, diseñado formalmente y fijado en un momento específico durante el ciclo de vida de ese elemento de configuración en este caso a veces nos referimos a una línea base con el nombre de “nueva versión”.

#### **4.3.mantenimiento correctivo**

Es la modificación de un producto software hecha después de la entrega debido a que debemos corregir errores descubiertos de modo que La modificación repara el producto software para satisfacer requerimientos.

#### **4.4.Plan de mantenibilidad**

Un documento que marca las practicas específicas del mantenimiento, así como los recursos y secuencia de actividades relevantes para el software de modo que el desarrollador es quien prepara éste plan.

#### **4.5.mejora**

Es un cambio software pero no es una corrección como las definidas antes así habrá dos tipos de mejoras: adaptativas y perfectivas.

#### **4.6.Plan de Mantenimiento**

Es un documento que dice cuales son las prácticas específicas del mantenimiento, los recursos y la secuencia de actividades relevantes para el mantenimiento de un software de modo que el mantenedor es quien prepara éste plan.

El plan debería ponerse en marcha una vez que el producto entre en la fase de mantenimiento.

#### **4.7.Proceso de mantenimiento**

El Proceso de Mantenimiento contiene las actividades y tareas que debe llevar a cabo el mantenedor.

Este proceso se activa cuando el producto software implica modificaciones en el

código y documentación asociada debido a un problema o la necesidad de mantenimiento para mejorar. El objetivo es modificar software existente preservando la integridad. Éste proceso incluye la migración y retiro del producto software.

#### **4.8.Programa de mantenimiento**

La estructura de la organización, responsabilidades, procedimientos, procesos y recursos usados para implementar el plan de mantenimiento de modo que el término "programa" significa lo mismo que "infraestructura".

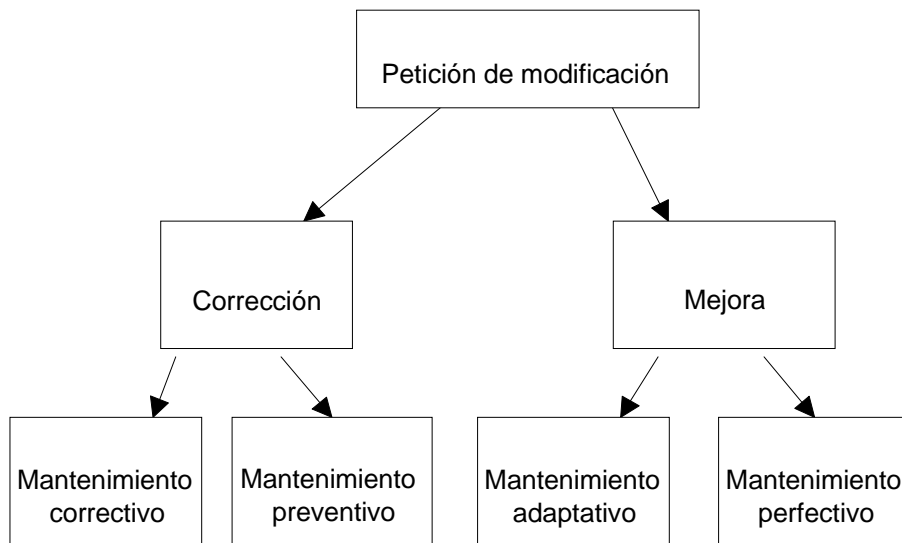
#### **4.9.Petición de Modificación (MR)**

Es un término genérico usado para identificar cambios de un producto software que se está manteniendo.

La MR puede clasificarse después como corrección o mejora e identificarla como correctiva, adaptativa, preventiva o perfectiva.

La MR también suele denominarse petición de cambio. En el dibujo siguiente podemos ver un esquema para clasificar los mantenimientos.

## Clasificación de peticiones de mantenimiento



### 4.10. Mantenimiento Perfectivo

La modificación de un producto software después de su entrega para mejorar el rendimiento o mantenibilidad.

El mantenimiento perfectivo proporciona mejoras para los usuarios, mejora de la documentación del programa, y recodificación para mejorar el rendimiento del software, su mantenibilidad u otros atributos.

### 4.11. Mantenimiento preventivo

Modificación del producto software tras la entrega para detectar y corregir fallos latentes antes de que se conviertan en fallos efectivos.

### 4.12. Informe de Problema

Término usado para identificar y describir problemas detectado en un software.

### 4.13. Entorno de Ingeniería del Software

El conjunto de herramientas, dispositivos firmware y hardware necesarios para llevar a cabo el trabajo de ingeniería del software.

Las herramientas pueden incluir compiladores, ensambladores, enlazadores, cargadores, sistemas operativos, depuradores, simuladores, emuladores, herramientas de prueba, de documentación y SGBD.



#### **4.14. Entorno de pruebas de software**

Las instalaciones, hardware, software, firmware, procedimientos y documentación necesarios para la cualificación y otras pruebas de software.

Se pueden incluir simuladores, analizadores de código, generadores de casos de prueba, analizadores de rutas, además de todos los incluidos en el entorno de Ingeniería del software.

#### **4.15. Transición del software**

Una secuencia coordinada y controlada de acciones donde el desarrollo del software pasa de desarrollo software inicial llevado a cabo dentro de la organización a mantenimiento de software llevado a cabo dentro de la organización.

### **5. Aplicación de éste estándar internacional**

En este punto se presenta el proceso de mantenimiento requerido para mantener productos software.

#### **5.1. Proceso de mantenimiento**

El Mantenimiento de Software es una de los cinco procesos primarios del ciclo de vida que se deben llevar a cabo durante el ciclo de vida del software (ISO/IEC 12207).

Los procesos primarios Adquisición y Entrega pueden iniciar la actividad Implementación del Proceso del proceso primario Mantenimiento de Software por medio de un acuerdo o contrato.

El proceso primario Operación de ISO/IEC 12207 pueden iniciar el proceso de Mantenimiento Software remitiendo una Solicitud de Modificación (MR) o Informe de Problema.

El proceso Mantenimiento de Software invoca el proceso primario Desarrollo de ISO/IEC 12207.

Los procesos de apoyo de Documentación, Gestión de la Configuración, Aseguramiento de la Calidad, Verificación, Validación, Revisión Conjunta, Auditoría y Resolución de Problemas de ISO/IEC 12207 se utilizan el proceso Mantenimiento de Software.

Los procesos del ciclo de vida relativos a la empresa de ISO/IEC 12207 constan de cuatro procesos.

La Gestión, Infraestructura, y Entrenamiento son procesos que emplea el mantenedor cuando inicia un proyecto de mantenimiento.

El Proceso de Mejora de ISO/IEC 12207 se enfoca para efectuar la mejora del proceso de mantenimiento de modo que La adaptación es apropiada para eventos no rutinarios como mantenimiento de emergencia y además de eso la adaptación de éste estándar internacional se describe en ISO/IEC 12207.

## **5.2.Organización de éste estándar internacional**

A continuación se presentan por orden los pasos que el mantenedor debería dar.

La sección 6 proporciona cosas a tener en cuenta en la Implementación, así como aspectos a considerar cuando se haga la planificación de mantenimiento.

La sección 7 proporciona información comprensible para la planificación.

La sección 8 cubre los detalles del Proceso de Mantenimiento incluyendo las tareas y los pasos dentro de cada una que se deben dar para implementar el Proceso de Mantenimiento.

## **6.Consideraciones sobre la Implementación**

### **6.1.Introducción**

El proceso del ciclo de vida Mantenimiento de Software empieza con la implementación de éste proceso donde se planifica el mantenimiento y acaba con la retirada del producto. Incluye la modificación de código y documentación debido a algún problema o la necesidad de mantenimiento.

El objetivo del Proceso de Mantenimiento es modificar un producto software existente preservando su integridad.

A continuación se dan algunas consideraciones sobre la implementación (del proceso de mantenimiento, no del software).

El Proceso de Mantenimiento es necesario ya que los productos software sufren cambios durante el ciclo de vida. Si el producto software se desarrolla usando herramientas CASE, todavía seguirá necesitando el mantenimiento. Éstas herramientas CASE facilitan el mantenimiento pero no lo eliminan. Si no se desarrolla código de aplicación, también hará falta el mantenimiento. El mantenimiento de productos de éste tipo, normalmente conlleva modificación de los interfaces, de los datos o de las operaciones que realiza.

Se deberían tener en cuenta los requerimientos implícitos y las restricciones impuestas sobre el desarrollador original. Las circunstancias pueden haber cambiado y puede que los requerimientos originales ya no sean válidos.

Durante la ejecución de los Procesos de Desarrollo, Operación y Mantenimiento de ISO/IEC 12207 cualquier problema detectado se graba y se sigue hasta el Proceso de Resolución de ISO/IEC 12207. Se envían solicitudes de Modificación o Informes de Problemas, llamados a menudo solicitudes de cambio.

El Proceso de Resolución de Problemas de ISO/IEC 12207 registra e informa el estado de las solicitudes de modificación o de los Informes de Problemas. También determina si alguna MR/PR intenta pedir una mejora.

El Proceso de Gestión de la Configuración de ISO/IEC 12207 registra e informa del estado de los MR/PR. La actividad de control de éste proceso llamada Control de la configuración decide si la solicitud se aprueba. Las MR/PR aprobadas se implementan llamando al Proceso de Mantenimiento.

El mantenimiento es necesario independientemente del modelo de ciclo de vida o del enfoque usado en el desarrollo.

El Proceso de Mantenimiento puede consumir una gran parte de los costes durante el ciclo de vida. El análisis de los tipos de mantenimiento puede ayudar a entender éstos costes.

## **6.2. Tipos de mantenimiento**

El mantenimiento Correctivo se refiere a los cambios necesarios debidos a algún error real en el software. Si el software no cumple los requerimientos debe hacerse éste mantenimiento.

El Preventivo se refiere a los cambios efectuados debido a la detección de posibles errores en el software. Se lleva a cabo en software que debe efectuar tareas de seguridad o de prevención de peligros para las personas.

El Adaptativo y Perfectivo son mejoras del software. Éstos cambios no estaban en las especificaciones de diseño del software entregado.

Los cambios Adaptativos son los necesarios para acomodar el producto a un entorno cambiante.

Se incluyen los cambios para implementar nuevos requerimientos de interfaz, de sistema o de hardware.

Los cambios Perfectivos mejoran el software, el rendimiento o la mantenibilidad.

Un cambio perfectivo puede proveer nuevas funcionalidades para los usuarios o para que la ingeniería inversa pueda crear documentación de mantenimiento que no existía antes.

El mantenimiento de software requiere hacer cambios en una estructura o sistema existente. Así las mejoras en forma de cambios adaptativos y perfectivos son muy caras.

Las mejoras pueden suponer una parte grande de los costes de mantenimiento y además se puede observar que distinguimos entre costes del ciclo de vida y costes de mantenimiento.

### 6.3. Gestión del mantenimiento

El comprador puede llegar a un acuerdo con el desarrollador original para ejecutar el mantenimiento o con una tercera empresa.

El mantenimiento también lo pueden llevar a cabo dos empresas en acuerdo.

La norma internacional ISO/IEC 12207 proporciona las tareas detalladas para llegar a un acuerdo entre comprador y vendedor. Esto podría utilizarse para llegar a un acuerdo si comprador y vendedor son de la misma organización o no.

Los aspectos específicos del mantenimiento se discuten más adelante.

El contrato debería dejar claro si el comprador solicita el mantenimiento al desarrollador después de la entrega o después del período de garantía.

Debería indicarse en el acuerdo si hay que entregar la documentación actualizada, además de si el contrato incluye cursos de entrenamiento.

El vendedor debería preparar procedimientos para el mantenimiento, mantener al día éstos procedimientos y comprobar que las actividades cumplen el acuerdo.

Los datos empíricos sugieren que el uso de procedimientos es muy caro. Los ítems a mantener, los procedimientos de mantenimiento y el tiempo durante el cual vamos a mantenerlos deberían especificarse en el plan de mantenimiento.

El vendedor y el comprador deberían primero llegar a un acuerdo sobre el mantenimiento y estipular los procedimientos a incorporar dentro del producto software.

Procedimientos similares se deberían usar por parte del desarrollador original y otros encargados del mantenimiento.

Estos procedimientos deberían incluir:

- Las reglas básicas usadas para determinar cuando el software debe ser corregido o cuando hace falta una nueva línea temporal (es decir el error es muy complejo) en la que usar el Proceso de Desarrollo de ISO/IEC 12207. Por ejemplo en una línea de código podemos equivocarnos al teclear y poner "<" en vez de "<=" y hacer que un bucle se deje un caso, ésto sería corregir un error. De forma contraria podemos equivocarnos al hacer el diseño de un algoritmo y entonces tener que gastar mucho más tiempo y dinero en hacer un nuevo diseño, volver a codificarlo...
- Descripciones de los tipos de versiones dependiendo de la frecuencia con que se hacen o de los efectos en el software. (este punto y los siguientes los comentaremos mas adelante).
- Formas de informar al comprador sobre el estado del proyecto.
- Métodos para confirmar que un cambio no provocará un nuevo error.
- Clasificación de los tipos de cambio, urgencia y su interrelación

## 6.4.Herramientas para el mantenimiento

Una posible forma de contener los costes del mantenimiento es el uso de herramientas CASE. Éstas herramientas ayudan en las actividades de mantenimiento.

Por CASE entendemos un conjunto de herramientas que ayudan en todos los aspectos del desarrollo y mantenimiento de software (ISO/IEC DTR 14471). Esta colección interrelacionada de herramientas CASE deberían formar un SEE (En inglés Entorno para la Ingeniería del Software) para soportar los métodos, políticas, líneas guía y estándares que soportan las actividades de mantenimiento de software.

Debería existir también un Entorno de Pruebas de Software (en inglés STE) de forma que el producto software modificado se pueda probar en un entorno no operativo.

El SEE proporciona las herramientas para comenzar a desarrollar y modificar los productos software. El STE proporciona el entorno para las pruebas.

Poner una fecha límite para el uso de herramientas CASE tiene que ver con el éxito en cierto modo los encargados del mantenimiento deberían planear estos trabajos con cuidado (ISO/IEC 14471).

## 6.5.Medida del software

La calidad del software es una consideración importante en el mantenimiento de un producto software.

Para la gestión de proyectos el modelo PMI a la gestión de calidad lo analiza (de modo que hay que asegurar que el proyecto satisface los requisitos, o sea las necesidades por los cuales fue emprendido) estudiando tanto la planificación, el aseguramiento como el control de la calidad siguiendo la trayectoria desde entradas a salidas:

- Planificación de la calidad; aplicando las técnicas del análisis de coste/beneficio, comprobación del rendimiento usando métricas, diseño de experimentos y construcción de diagramas de flujo a la política de calidad y a la descripción del producto obtendremos un plan para la gestión de la calidad y unas listas de comprobación.

- Aseguramiento de calidad: este concepto proviene de la mentalidad empresarial japonesa (“kai-zen”) y en pocos años se ha hecho muy popular hasta el punto de estar incluido en este estándar internacional. En realidad “kai-zen” incluye muchas cosas mas que el aseguramiento de la calidad, desde por ejemplo garantías de productos hasta políticas de empleo (en Japón hasta hace pocos años la relación entre el trabajador y la empresa duraba toda la vida laboral de forma que en algunas zonas era tradicional que los hijos heredasen el puesto del padre). Conseguimos mejoras de calidad al utilizar técnicas de auditoria sobre el plan de gestión de calidad basándonos en los resultados de las

métricas de calidad.

- Control de calidad: este proceso recibe como entradas el plan de gestión de calidad y las listas de comprobación y utilizando técnicas estadísticas logramos mejoras en la calidad así como un mejor ajuste de los procesos.

Los encargados del mantenimiento deberían tener un programa de calidad de software que incluya las seis características de la calidad software descritas en ISO/IEC 9126.

Se debería implementar un proceso que identifique, defina, seleccione, aplique, valide y mejore la medida del software con vistas al mantenimiento.

Como parte de la medida del software, el encargado de mantenimiento debería determinar el esfuerzo (en términos de recursos gastados) para el mantenimiento perfectivo, adaptativo, preventivo y correctivo.

Los datos se deberían recoger, analizar e interpretar para facilitar la mejora del Proceso de Mantenimiento y para obtener un mejor entendimiento de donde se gasta más dinero en el mantenimiento.

Se deberían recoger métricas empíricas para ayudar a estimar el coste del ciclo de vida.

## **6.6.Documentación del proceso**

El proceso de Mantenimiento detallado (ver capítulo 8) debería documentarse de forma que todo el personal de mantenimiento siga el mismo proceso. Las métricas deberían apoyar el proceso y los esfuerzos para la mejora del proceso software.

## **6.7.Involucramiento temprano en el desarrollo**

Los datos sugieren que el coste del mantenimiento y la habilidad del personal de mantenimiento para dirigir el mantenimiento está muy influenciado por lo que pasa o no pasa durante el proceso de desarrollo.

En muchos casos, el personal de mantenimiento no puede involucrarse debido a los contratos o a otras razones. En el caso específico de que el mantenimiento se encargue a una tercera compañía a menudo no hay forma de involucrarse. Cuando el personal de mantenimiento pueda intervenir durante el desarrollo deberá hacerlo.

Entre las funciones que desarrolla el encargado del mantenimiento deberíamos incluir:

- Plan para la estrategia de soporte al producto. Esto determina que clase de servicios, de que tipo y de que forma los vamos a ofrecer.
- Asegurar que el producto puede recibir soporte. Por ejemplo una pequeña empresa no puede dar soporte a clientes ubicados en el extranjero.

- Dar soporte a la planificación de la transición de desarrollo a mantenimiento. Debemos hacer un plan para esta transición porque es más cara y dura más tiempo. Por ejemplo las fases de análisis y diseño solo se dan una vez mientras que la implementación puede darse cada vez que se pida mantenimiento.

La planificación se discute en detalle en el capítulo 7.

La facilidad de dar soporte a un producto software incluye tareas tales como probar y asegurar la mantenibilidad. ISO/IEC 9126 maneja la mantenibilidad y otras características que son importantes durante el desarrollo.

La facilidad para dar soporte se puede mejorar mediante la participación del encargado de mantenimiento en el aseguramiento de la calidad, verificación y validación que dan soporte a los procesos del ciclo de vida de ISO/IEC 12207.

El encargado de mantenimiento debería:

- Participar en las revisiones.
- Realizar análisis del código.
- Hacer un estudio continuo de los requerimientos.
- Ejecutar la Verificación y Validación.

Estos puntos se comentarán mas abajo.

## **6.8.Mantenibilidad**

La mantenibilidad del software y el mantenimiento son aspectos importantes en cuanto a su dependencia.

La mantenibilidad es una característica importante para el comprador, el vendedor y el usuario. De modo que podremos definir, verificar Los requerimientos de mantenibilidad deberían incluirse en la Actividad de Iniciación del Proceso de Adquisición de ISO/IEC 12207 y debería evaluarse durante el Proceso de Desarrollo de ISO/IEC 12207.

Las variaciones en el diseño deberían ser estudiadas en todo momento durante el desarrollo para ver el impacto en la mantenibilidad.

Se deberían usar varias medidas para poder definir y calcular la calidad del software. La evaluación tanto cualitativa como cuantitativa es importante.

La mantenibilidad es una característica de la calidad del software que afecta a la velocidad y facilidad de cambios en el software antes de su entrega para su uso. (ISO/IEC 9126).

### **6.8.1.Mantenibilidad y el proceso de desarrollo**

La mantenibilidad debería tenerse en cuenta antes de empezar a

desarrollar. Deberían desarrollarse acuerdos entre el comprador y el vendedor como parte de la Actividad Iniciación del proyecto de ISO/IEC 12207.

El desarrollador debería preparar un Plan de Mantenibilidad que proporcione prácticas específicas de mantenibilidad, así como recursos y secuencias relevantes de actividades. También debería establecerse el esfuerzo necesario para dar soporte al seguimiento y evaluación de aspectos de la mantenibilidad del Proceso de Desarrollo de ISO/IEC 12207.

La capacidad para seguir y evaluar aspectos sobre la mantenibilidad se debería desarrollar durante el desarrollo de software.

La capacidad describe los requerimientos de mantenibilidad del software de forma cualitativa y cuantitativa. Define los criterios y formas de comprobarlos.

Los requerimientos cualitativos se usan para definir las técnicas empleadas para facilitar el mantenimiento en cuanto a costes y recursos.

Los requerimientos cuantitativos se usan para definir las magnitudes de mantenibilidad o criterios de calidad así como las métricas usadas para determinar valores clave durante las distintas fases del ciclo de vida.

La efectividad de éste esfuerzo durante el desarrollo se hace evidente una vez que las actividades de mantenimiento comienzan. Los desarrolladores deberían implementar los requerimientos para la mantenibilidad y los encargados del mantenimiento deberían seguir la pista a la implementación. Éste esfuerzo debería ser parte de la estrategia de mantenimiento del software.

Uno de los factores clave en la aplicación de ISO/IEC 12207 es el desarrollo de una estrategia de mantenimiento de software (ver Guía de ISO/IEC TR 15271). De acuerdo con esto se debería desarrollar una estrategia de mantenimiento además de planear el mantenimiento (ver capítulo 7).

Antes del diseño se debería establecer una estrategia de mantenimiento de software. El hacerlo tan temprano permitirá al encargado del mantenimiento ahorrar dinero. Hay muchas acciones, incluyendo la planificación del mantenimiento software, que se deben llevar a cabo durante el proceso de desarrollo. Éstas acciones se deberían documentar en el plan de mantenimiento de software (ver sección 7.3.2).

Los siguientes aspectos, que afectan a la mantenibilidad, deberían tenerse en cuenta al elegir el lenguaje de programación:

- Portabilidad del lenguaje.
- Legibilidad del lenguaje.
- Estabilidad del lenguaje.
- Auto-documentación. Por ejemplo hoy en día se ha hecho muy famoso la herramienta “javadoc” que genera automáticamente la documentación asociada al programa.
- Tolerancia a “trucos” que reducen la claridad.



- Posibilidades de estructuración. Intentar usar lenguajes que ofrecen facilidades para la estructuración, tal como los lenguajes orientados a objetos como Java. En el peor de los casos tenemos lenguajes como ensamblador que no ayudan a la creación de módulos.
- Facilidad para el desarrollo de nuevas versiones.
- Posibilidades para la estructuración de datos. Por ejemplo hay lenguajes como las primeras versiones de Basic que no permitían crear registros.
- Existencia de compiladores. Por ejemplo puede ocurrir que se usen lenguajes cuyos compiladores sean de coste alto.
- Estabilidad del compilador. Ni el compilador ni su código se deben colgar.
- Posibilidades para la comprobación durante la compilación y ejecución. Por ejemplo en el compilador Borland Pascal se puede depurar el código paso a paso.
- Existencia de entornos para Ingeniería del Software y para Prueba del Software para ayudar en la producción, depuración y gestión de la configuración y satisfacción de requerimientos de calidad y fiabilidad.
- Vida de las herramientas de desarrollo.

## **6.8.2.Mantenibilidad y actividades específicas del proceso de desarrollo**

### **6.8.2.1.Análisis de requerimientos software**

Las especificaciones software deberían describir de forma exacta y sin ambigüedades los requerimientos de mantenibilidad del software.

Éstos requerimientos se deberían incluir en las especificaciones de características de calidad requeridas por ISO/IEC 12207.

Los siguientes aspectos afectan a la mantenibilidad y deberían tenerse en cuenta:

- Identificación y definición de funciones del programa, sobre todo las opcionales.
- Exactitud y organización lógica de los datos, por ejemplo poner las bases de datos en forma normal

Interfaces, es decir la especificación de las interfaces de usuario.

- Requerimientos de rendimiento, por ejemplo “el sistema debe tardar menos de x segundos”, incluyendo los efectos de correcciones y añadidas, por ejemplo “si añadimos la función F el sistema se retarda en Y segundos”.
- Requerimientos impuestos por el entorno planificado, por ejemplo no gastar mas dinero del presupuestado. En resumen ajustarse al gasto de recursos previsto por la planificación
- Granularidad de los requerimientos ya que esto afecta a la dificultad o trazabilidad, se puede traducir por “obtener el mayor grado de detalle en el análisis de requerimientos”.
- El Plan de Aseguramiento de la calidad del software debería poner énfasis en la documentación y su cumplimiento, debe poner bien claro como documentar el programa y además se debe garantizar su cumplimiento.

### **6.8.2.2.Diseño de la arquitectura del software**

Ésta actividad transforma los requerimientos del software en una arquitectura que describe su estructura de alto nivel y que identifica los componentes software (ISO/IEC 12207).

Las principales características de ésta Actividad del Proceso de Desarrollo de ISO/IEC 12207 que afectan a la mantenibilidad son la elección de la estructura del programa, su división en entidades y el flujo de datos entre ellos. Como en otras actividades, es importante usar los conocimientos sobre procesamiento de datos que tenga el equipo ya que esto puede revelar posibilidades importantes sobre la reutilización de partes de programas existentes o bibliotecas de funciones que ya han demostrado su utilidad.

El diseño modular, combinado con el análisis top-down, y una adecuada documentación, que nos permitirá añadir y quitar cosas fácilmente, son las dos principales características que harán que se continúe cumpliendo los requerimientos de mantenibilidad.

### **6.8.2.3.Diseño detallado del software**

Ésta actividad del Proceso de Desarrollo de ISO/IEC 12207 proporciona un diseño detallado para cada componente software, para los interfaces y las bases de datos.

Ésta actividad produce una descripción exacta y detallada de las funciones necesarias para completar la solución de programación propuesta.

La mantenibilidad del software mejorará con la inclusión de características de calidad de ISO/IEC 9126.

#### **6.8.2.4.Codificación y pruebas del software**

Ésta Actividad del Proceso de Desarrollo de ISO/IEC 12207 desarrolla, documenta y prueba las unidades software, así como las bases de datos.

La mantenibilidad del software mejorará si vamos actualizando la documentación. Una documentación de calidad debería proporcionar información que ayude en el Proceso de Mantenimiento.

Algunas sugerencias para mejorar la mantenibilidad con una documentación de calidad:

- Garantizar la legibilidad, por ejemplo en un programa debe haber nombres significativos tanto para variables, constantes, tipos, funciones..., y para facilitar su entendimiento debe haber código bien comentado.
- Evitar código no estructurado, modularizar el programa al máximo.
- Examinar las debilidades del lenguaje para evitar problemas clásicos, por ejemplo en C tenemos la posibilidad de hacer preincrementos y postincrementos, supongamos que escribimos en el código esta expresión: `i++ + ++j` . El programador puede esperar un resultado de dicha expresión distinto del que se obtiene al ejecutar el programa.
- Detección de errores en el diseño detallado, si no se hace eso se puede llegar a perder tiempo.
- Uso de técnicas que faciliten el seguimiento de errores, por ejemplo trazar el programa.

#### **6.8.2.5.Pruebas de cualificación de software**

Ésta actividad asegura que la implementación de cada requerimiento software se comprueba para ver si se cumple dicho requerimiento (ISO/IEC 12207).

Los requerimientos software relacionados con la calidad se prueban durante ésta actividad.

Los casos de prueba usados durante el desarrollo de software deberían guardarse para hacer un análisis de regresión después de las modificaciones.

Además, la historia del desarrollo de un programa debería estar disponible para evitar repetir errores y contribuir al mantenimiento desde el principio.

## **6.9.Transición del software**

La transición del software es una secuencia controlada y coordinada de acciones donde el desarrollo pasa de la organización que ejecuta el desarrollo inicial a la organización encargada del mantenimiento.

Si la responsabilidad del mantenimiento se transfiere a una organización distinta se debería desarrollar un Plan de Transición.

El Plan debería tratar:

- La transferencia de hardware, software, datos y experiencia del desarrollador al encargado del mantenimiento, el que ha hecho el programa debe comunicar lo que sabe al encargado del mantenimiento.
- Tareas necesarias para que el encargado de mantenimiento pueda implementar una estrategia de mantenimiento de software (p. ej. composición de personal, entrenamiento, instalación y réplica de problemas de mantenimiento).

## **6.10.Documentación**

Los encargados del mantenimiento a menudo se enfrentan problemas tales como tener que proporcionar mantenimiento a un producto con poca o ninguna documentación.

Si no hay documentación el encargado debería crearla. La creación de documentación es parte del mantenimiento perfectivo. Ésto hace que se presenten dificultades en el proceso de mantenimiento.

Cuando nos enfrentemos a ésta situación deberíamos seguir los siguientes pasos para preparar el mantenimiento:

- Entender el dominio del problema (tipo de aplicación). Leer la documentación (si la hay), discutir sobre el producto con los desarrolladores (si es posible) y operar con el producto software.
- Conocer la estructura y organización del producto software. Hacer un inventario sobre él, producir árboles de llamadas y analizar su estructura.
- Determinar que hace el producto software. Revisar las especificaciones (si las hay), revisar la estructura general, analizar los árboles de llamadas, hacer presentaciones orales para el resto de

personal de mantenimiento y añadir comentarios al código.

- Corregir las prioridades bajas de propuestas de modificación o informes de problemas, por ejemplo puede que alguien otorgue una prioridad baja a un problema y en realidad sea un problema muy prioritario.

Los encargados del mantenimiento deberían documentar el software según los pasos anteriores.

A medida que se haga necesario se deben actualizar o crear documentos tales como especificaciones, manuales de mantenimiento para programadores, manuales de usuarios o guías para la instalación.

Hay varios factores que influyen en la creación/actualización del entorno de mantenimiento. Entre ellos contamos: acceso al código fuente, disponibilidad de herramientas para análisis de código, capacidad para operar con el producto para determinar capacidades y disponibilidad de un entorno para las pruebas.

## **7.Estrategia para el mantenimiento de software**

### **7.1.Introducción**

Aquí discutiremos el desarrollo de una estrategia la cual prepara los recursos humanos y materiales necesarios para proporcionar mantenimiento para productos software.

Los resultados de los análisis de mantenibilidad deberían usarse como ayudas en la planificación del mantenimiento.

Los análisis deberían usarse como una entrada en el desarrollo de la estrategia de mantenimiento.

La estrategia de mantenimiento de software debería consistir de los siguientes elementos:

- Concepto de mantenimiento
- Plan de mantenimiento
- Análisis de recursos

Estos puntos se van a discutir en los apartados siguientes.

### **7.2.El concepto de mantenimiento**

Determinar el concepto de mantenimiento debería ser el primer paso en el desarrollo de la estrategia para el mantenimiento de software.

El concepto de mantenimiento debería desarrollarse cuando las necesidades del producto software inicial se expresen.

El concepto de mantenimiento debería manejar:

- El alcance del mantenimiento del software
- La personalización del proceso
- La designación de la persona encargada del mantenimiento
- Una estimación de los costes de mantenimiento

El Concepto de Mantenimiento se documenta en el Plan de Mantenimiento.

### **7.2.1.Alcance**

El alcance determina la responsabilidad del mantenedor. Debería definir el soporte que el encargado del mantenimiento debe ofrecer. A menudo, las restricciones monetarias rigen el alcance del mantenimiento.

El alcance debería manejar:

- Tipos de mantenimiento a ejecutar, los ya consabidos adaptativo, correctivo, preventivo, perfectivo.
- Nivel de la documentación a mantener, si el proyecto es muy comprensible, es decir lleva mucha información entonces será fácil de mantenerlo.
- Responsabilidad, determinarla de forma clara.
- Nivel de entrenamiento que debemos dar, a alta complejidad del software alto nivel de entrenamiento.
- Soporte de la entrega, dejar claro si se va a dar soporte o no.
- Soporte de ayuda on-line, especificar bien si se va a dar soporte en línea o no.

### **7.2.2.Personalización del proceso**

El Concepto de Mantenimiento debería manejar las tareas de mantenimiento de software después de la entrega.

Las diferentes organizaciones pueden necesitar ejecutar distintas tareas durante el mantenimiento. Un intento al principio debería identificar éstas organizaciones y documentarlos en el Concepto de Mantenimiento.

El Concepto de Mantenimiento debería reflejar el Proceso de

Mantenimiento que empleamos.

### **7.2.3.Designación de la persona encargada del mantenimiento**

Ésto es algo importante y debería manejarse al principio y documentarlo en el Concepto de Mantenimiento. Ésto también se debe hacer aunque sea dentro de la misma organización.

Para acuerdos con terceras personas, el Concepto de Mantenimiento debería hacer notar que el mantenimiento será de ésta manera. El proceso primario de Adquisición y Abastecimiento de ISO/IEC 12207 proporciona detalles relacionados con los servicios de adquisición y abastecimiento.

La designación debería tener en cuenta éstos factores:

- Vida del producto software: la complejidad del proceso de producción de software se intenta abordar mediante la descomposición en diversas etapas, dicha descomposición define el ciclo de vida del producto software.
- Costes de iniciación: el mantenimiento es la parte más costosa del ciclo de vida del producto software. Aunque es menos costoso detectar y corregir un error durante las etapas de iniciación del ciclo de vida que durante las últimas.
- Disponibilidad de espacio: por ejemplo antes de modificar un producto software primero tiene que asegurar que hay espacio suficiente para guardar los cambios ya que si no se tiene en cuenta esto puede ocurrir que no se guarde todo el producto con todos los cambios realizados. Esto puede provocar un desorden en el código y pueden surgir varios errores al respecto.
- Cualificación: primero y ante todo se debe saber que cualidades tiene el producto software y que cualidades se quieren conseguir y estudiarlas, esto facilita el mantenimiento del software.
- Disponibilidad: cuando hay una disponibilidad de mantener un producto software es cuando se comienza el desarrollo y el estudio del mantenimiento del software.
- Planificación: es el que más influye positivamente en el mantenimiento del software. El producto está sujeto a cambios. Existen métricas para la mantenibilidad de esfuerzo (evalúa el esfuerzo requerido durante la fase de mantenimiento), de complejidad y de estructura (analiza la correlación entre la estructura de un programa y su facilidad de mantenimiento).
- Conocimiento del dominio del problema: se hace un análisis detallado del problema de modo que hay que dejar claro las partes que se van a modificar y las que no se van a modificar. Además se debe saber el motivo o el problema por lo cual se hace el cambio o la modificación esto facilita el mantenimiento del software.

#### **7.2.4. Estimación de los costes de mantenimiento.**

Debería prepararse una estimación de los costes de mantenimiento.

Asegurar que el proyecto es completado dentro del presupuesto previsto es lo que nos propone el modelo PMI, para ello hace un análisis de la planificación de recursos, estimación de costes, realización del presupuesto y control de costes, veámoslos desde el punto de las entradas hasta las salidas:

- Planificación de recursos; se obtiene un documento que especifica los requerimientos de recursos de nuestro proyecto. Para ello necesitaremos la opinión de expertos y algunas alternativas (por si surgen problemas con los recursos). Nuestro plan parte de una estructura de descomposición de trabajos y una declaración de objetivos y si la hay de información histórica.
- Estimación de costes; se pretende conseguir una estimación detallada además de un plan de gestión de costes. Partimos de una estructura de descomposición de trabajos, unos requerimientos de recursos, estimaciones sobre la duración de las actividades y información histórica. Las técnicas que usaremos serán: modelado de parámetros de interés para el problema, estimación de menor a mayor detalle y en problemas de tamaño grande será recomendable usar herramientas informáticas.
- Realizar presupuesto de costes; las entradas a este proceso son: estimación de costes, la estructura de descomposición de trabajos y la salida es una línea de base para el coste. Las técnicas a utilizar variarán según el problema.
- Control de costes; este punto es bastante amplio, de hecho se debe tener en cuenta varias entradas, a saber: línea de base para el coste (obtenidas en el punto anterior), informes de rendimiento, informes sobre cambios y el plan de gestión de costes. Aplicando a lo dicho anteriormente las siguientes técnicas: sistemas de control de cambios en el coste, métricas de rendimiento y herramientas automáticas se obtienen: estimaciones de coste revisadas, presupuestos actualizados, acciones correctivas y lecciones para el futuro.

Los costes deberían ser función del alcance del mantenimiento.

Factores adicionales a tener en cuenta serían:

- Viajes a las ciudades de los usuarios: estos proporciona un coste el cual se debe incluir en los costes del mantenimiento del software.
- Entrenamiento de encargados de mantenimiento y usuarios: esto requiere un tiempo donde se gasta dinero en el



aprendizaje y el entrenamiento del personal o el grupo del mantenimiento.

- Costes y mantenimiento anual de los entornos de Pruebas y de Ingeniería del Software: antes de hacer cualquier cambio en un producto software se realizan pruebas, esto es para obtener un mantenimiento seguro y fiable, esto requiere tiempo y personal que se encarga de hacer dichas pruebas, y esto a su vez induce a nuevos costes que se añaden al los costes del mantenimiento del software.
- Costes de personal, como salarios y beneficios.

Cuando se desarrolla el Concepto de Mantenimiento, los costes se deberían estimar basándonos en los limitados datos disponibles. A medida que el desarrollo progresa las estimaciones deberían refinarse.

Las métricas históricas deberían usarse como entradas para estimar los costes de mantenimiento.

## **7.3. Planificación del mantenimiento**

### **7.3.1. Introducción**

El objetivo es planificar las actividades de mantenimiento y adquirir los recursos necesarios lo antes posible para que estén disponibles tan pronto como el producto software tenga que mantenerse.

La planificación se inicia una vez que el concepto de mantenimiento de software se haya definido y culmine en un plan de mantenimiento usado para guiar a los encargados de mantenimiento cuando el producto entre al servicio del cliente.

### **7.3.2. El plan de mantenimiento**

La planificación de las actividades y tareas de mantenimiento deberían empezar tan pronto como el Concepto de Mantenimiento se defina. Culmina en la preparación de un Plan de Mantenimiento. El Plan de Mantenimiento debería prepararse durante el desarrollo de software por el mantenedor y debería incluir la forma que tienen los usuarios de solicitar cambios al producto software.

El Plan de Mantenimiento debería cubrir

- Porqué hará falta el mantenimiento.
- Quién hará ese trabajo.
- Papeles y responsabilidades de las personas involucradas: a cada persona se le asigna la tarea que le corresponde y asume cierta responsabilidad sobre la resolución y el estudio de dicha

tarea.

- Como se hará el trabajo: sé hacer un diseño y un plan a seguir durante el mantenimiento
- Qué recursos habrá disponibles para el mantenimiento.
- Donde se hará el mantenimiento.
- Cuando comenzará el mantenimiento.

### **7.3.3.Guías para el plan de mantenimiento**

Aquí desarrollamos las guías para desarrollar el plan de Mantenimiento. Se incluyen temas clásicos para su inclusión en un Plan de Mantenimiento. Basado en el tamaño del esfuerzo se debería tomar una decisión sobre que aspectos incluir:

#### **1.Introducción**

- Describir el sistema al que debemos dar soporte: se especifican todos los detalles del sistema a mantener
- Identificar el estado inicial del software: eso para saber cuales son los cambios nuevos realizados
- Describir que soporte es necesario: esto para facilitar el comienzo del desarrollo del mantenimiento del software
- Identificar la organización que debe hacer el soporte/mantenimiento: para contemplar el objetivo del mantenimiento en el proceso de desarrollo del software.
- Describir cualquier acuerdo entre cliente y vendedor: Se debe tener claro lo que quiere el cliente pero por escrito de este modo el vendedor sabe lo que es lo que tiene que hacer para satisfacer el cliente.

#### **2.Concepto de mantenimiento**

Para definir el concepto del mantenimiento debemos definir o saber los siguientes puntos:

- Describir el concepto
- Describir el nivel de soporte para el sistema: Desde donde y hasta donde le vamos dar soporte al sistema.
- Identificar el período de soporte: cuanto tiempo se va a tardar en dar el soporte al sistema
- Personalizar el proceso: Se pone a medida de las necesidades.

### 3.Actividades de mantenimiento y de la organización

En este caso primero vamos a especificar el papel y la responsabilidad del mantenedor antes de la entrega:

- Implementación del Proceso
- Establecer la Infraestructura
- Establecer el Proceso de Entrenamiento
- Establecer el Proceso de Mantenimiento

En segundo lugar definimos el Papel y responsabilidad del mantenedor después de la entrega:

- Implementación del Proceso. Debido a su complejidad le dedicamos un punto aparte mas adelante.
- Análisis del Problema y de la Modificación
- Implementación de la Modificación
- Revisión/Aceptación del Mantenimiento
- Migración
- Retiro
- Resolución de Problemas (lo que incluye ayuda on-line)
- Entrenar al Personal

Por último definimos el papel del usuario

- Pruebas de aceptación
- Interfaz con otras organizaciones

### 4.Recursos

Se trata de los recursos humanos que participan en el proyecto, en este caso se define el tamaño del equipo de proyecto.

Hay que identificar tanto el software como el hardware necesario para el sistema de soporte (incluyendo sistema más requerimientos de herramientas STE/SEE).

Respecto a las instalaciones mas bien se debe identificar el hardware, y a la documentación hay que saber cual es el plan de calidad del software, el de gestión de proyecto, el de gestión de configuración, también de debe identificar los documentos de

desarrollo, y los manuales para el mantenimiento.

La verificación del plan es importante tanto como su validación de modo que hay que hacer un plan de pruebas, pruebas de procedimientos, informes sobre pruebas, y de entrenamiento.

Para facilitar el entendimiento ha de tener un manual de usuarios.

Hay que identificar que datos de se van a obtener y que datos se van a usar en el proyecto eso da lugar a la facilidad de seguimiento durante todo el ciclo de desarrollo de dicho proyecto de modo hay que ver si existen otros requerimientos (si los hay).

5. Proceso (como se va a llevar a cabo el trabajo)

- Proceso del encargado del mantenimiento (dar una visión global del proceso, no describir el proceso completo)
- Proceso personalizado

6. Entrenamiento

- Identificar necesidades de entrenamiento de Mantenedores y Usuarios

7. Registros e informes de mantenimiento

- Listas de peticiones de ayuda, modificación o informes de problemas
- Estado de las peticiones (ordenado por categorías)
- Prioridades de las peticiones
- Métricas a recoger en las actividades de mantenimiento

## **7.4 Análisis de recursos**

El último elemento de una estrategia de mantenimiento de software es el análisis de recursos. Una vez el alcance de mantenimiento y quien lo va a hacer se conozcan, el personal, el entorno de mantenimiento y los requerimientos de recursos financieros pueden conocerse.

El comprador, con ayuda del vendedor (desarrollador) normalmente determina los requerimientos de recursos para el mantenimiento software. Se deberían manejar el personal, entorno, y recursos financieros.

### **7.4.1 Recursos de personal**

Uno de los principales aspectos en la planificación del mantenimiento software es la planificación de requerimientos de recursos para el mantenimiento de software. Los requerimientos de personal son un factor de coste importante, y a la vez, el más difícil de determinar exactamente. Los dos enfoques más populares para estimar los recursos es el uso de modelos paramétricos y el uso de experiencia.

Los modelos requieren datos empíricos históricos. El mejor enfoque al usar la experiencia es tener datos históricos empíricos.

Se sugiere que se use una metodología estándar para la estimación de mantenimiento basada en el acuerdo. Se debería desarrollar un estudio separado del personal de mantenimiento que maneje la metodología para determinar los recursos de personal y los resultados.

#### **7.4.2 Recursos del entorno**

El desarrollo y mantenimiento de software son actividades especializadas y necesitan sistemas separados y dedicados.

Los Entornos para la Ingeniería del Software y para las Pruebas del Software deberían estar separados. El encargado del mantenimiento debería ayudar al comprador con el plan para el entorno de mantenimiento.

Conseguir que el entorno de mantenimiento se incluya en la planificación inicial es algo crítico cuando se asignan los fondos y se determina un presupuesto para el desarrollo y mantenimiento del producto software

#### **7.4.3 Recursos financieros**

El tercero y último aspecto de los recursos es el de los recursos financieros. Para proporcionar apoyo de mantenimiento el mantenedor debería tener un presupuesto que maneje los siguientes aspectos:

- Salarios, incluyendo las horas extras que se hagan falta.
- Entrenamiento (2-3 semanas por persona y año)
- Costes anuales de mantenimiento para licencias de software
- Viajes
- Publicaciones técnicas en forma de libros o revistas.
- Hardware y software necesarios para los entornos de ingeniería y pruebas.
- Actualizaciones de los anteriores, puede resultar que a costa de una actualización de un producto software hará falta una actualización de hardware costosa.

## 8. Los procesos de Mantenimiento

Aquí definimos las actividades y tareas para el proceso primario del ciclo de vida llamado "mantenimiento software".

El Proceso de Mantenimiento contiene las actividades y tareas necesarias para modificar un producto software existente conservando su integridad. Éstas actividades y tareas son responsabilidad del encargado del mantenimiento. Éste estándar internacional proporciona los pasos de cada tarea, que son ejemplo de que hacer para implementar las actividades y tareas de mantenimiento. El encargado del mantenimiento debería asegurarse de que el Proceso de Mantenimiento existe y funciona antes de desarrollar un producto software. El Proceso de Mantenimiento debería activarse cuando existe un requerimiento de mantenimiento de un producto software.

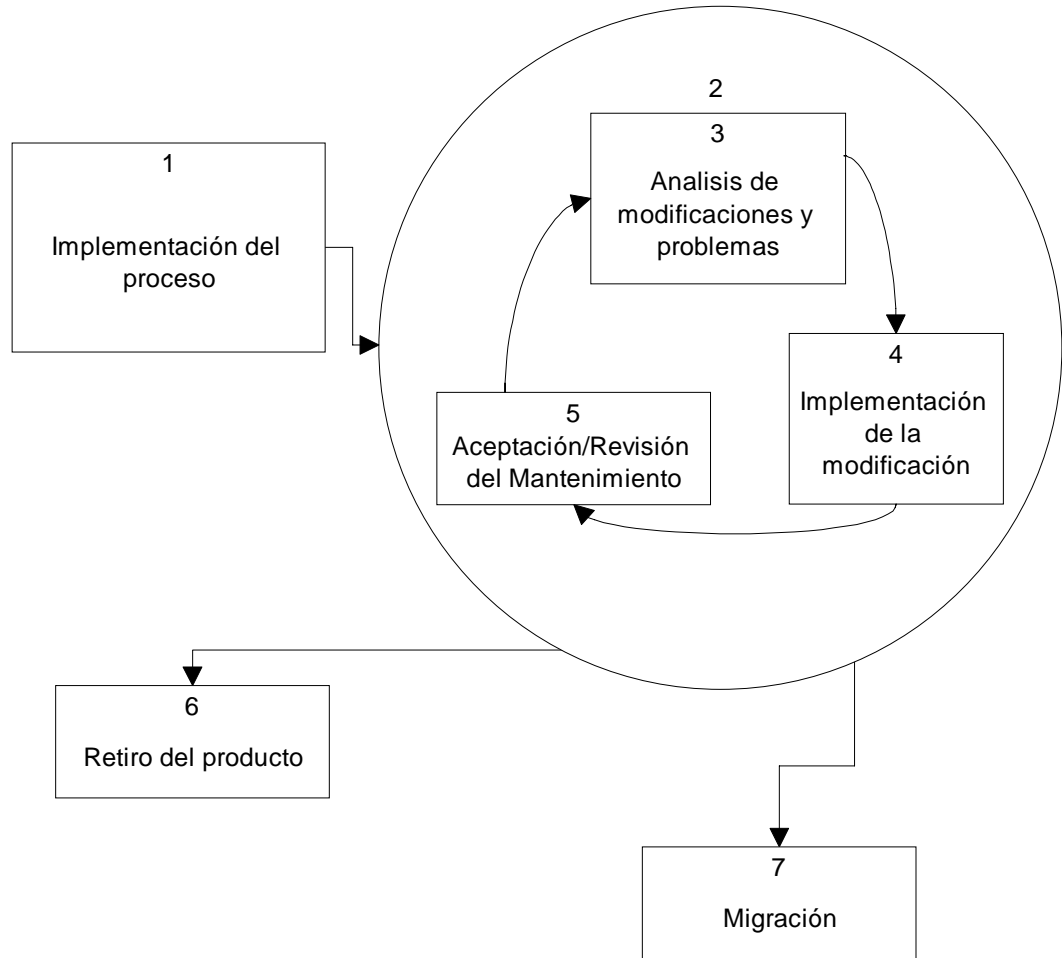
Tan pronto como se active el proceso, se deberían desarrollar los Planes y Procedimientos de Mantenimiento y se deberían asignar los recursos para el mantenimiento. Después de que el producto software se entregue, los encargados del mantenimiento deberían modificar el código y documentación asociados como respuesta a una petición de modificación o informe de problema. El objetivo global del mantenimiento software es modificar el producto software conservando su integridad.

Éste proceso da soporte al producto software desde su nacimiento pasando por la migración a otro entorno hasta su retiro. El proceso finaliza cuando el producto software es retirado.

Las actividades que comprende el Proceso de Mantenimiento son:

- Implementación del Proceso
- Análisis de Modificaciones y Problemas
- Implementación de Modificaciones
- Revisión/Aceptación del Mantenimiento
- Migración
- Retiro

En el gráfico siguiente podemos ver una representación de la secuencia de actividades que conlleva el mantenimiento.



Las entradas se transforman o consumen por las actividades de mantenimiento para producir salidas. Los controles proporcionan una guía para asegurar que la actividad de mantenimiento produce salidas correctas.

Las salidas son los objetos o datos producidos por la actividad de mantenimiento. Lo que llamamos "dar soporte" identifica los procesos del ciclo de vida relacionados con la organización y el soporte de ISO/IEC 12207 usados por las actividades de mantenimiento.

## 8.1. Implementación del Proceso

Durante la Implementación del Proceso, el encargado del mantenimiento establece los planes y procedimientos a ejecutar durante el Proceso de Mantenimiento. El Plan de Mantenimiento debería desarrollarse en paralelo con el Plan de Desarrollo. El encargado del mantenimiento debería establecer también los interfaces con la organización necesarios durante ésta actividad. Al hablar de interfaces nos referimos a las formas de comunicación que utilizaremos con la empresa que nos solicita el mantenimiento.

### 8.1.1 Entradas

Deberíamos incluir:

- La Línea seguida anteriormente

- Documentación del Sistema
- Una solicitud de modificación (MR) o informe de problema (PR)

## **8.1.2 Tareas**

Para implementar de forma efectiva el Proceso de Mantenimiento, el encargado de mantenimiento debería desarrollar y documentar una estrategia para ejecutar el mantenimiento. Para conseguir ésto, se deberían llevar a cabo éstas tareas:

- Desarrollar Planes y Procedimientos de Mantenimiento
- Establecer procedimientos para MR/PR
- Implementar la gestión de la configuración

### **8.1.2.1 Planes y procedimientos de mantenimiento**

El encargado de mantenimiento deberá desarrollar, documentar y ejecutar los planes y procedimientos para dirigir las actividades y tareas del Proceso de Mantenimiento

El Plan de Mantenimiento debería documentar la estrategia a usar para mantener el sistema, mientras que los Procedimientos de Mantenimiento deberían proporcionar un enfoque más detallado para conseguir efectuar el mantenimiento. Para desarrollar Procedimiento y Planes de Mantenimiento efectivos, el encargado del mantenimiento debería llevar a cabo los siguientes pasos:

- Ayudar al comprador a desarrollar el concepto de mantenimiento
- Ayudar al comprador a determinar el alcance del mantenimiento
- Ayudar al comprador a analizar alternativas a la organización del mantenimiento
- Garantizar una designación por escrito como encargado del mantenimiento del producto
- Dirigir los análisis de recursos
- Estimar los costes de mantenimiento
- Hacer un cálculo de la mantenibilidad del sistema
- Determinar los requerimientos para la transición



- Identificar los procesos de mantenimiento a usar
- Documentar el proceso de mantenimiento en forma de procedimientos operativos

### **8.1.2.2 Procedimientos para las peticiones de modificación**

El mantenedor debería establecer (ver norma ISO/IEC 12207 punto 5.5.1.2) procedimientos para la recepción, grabación y seguimiento de los informes de problemas y peticiones de modificación de los usuarios y proporcionar a los usuarios la realimentación necesaria.

El mantenedor debería llevar a cabo los siguientes pasos/tareas:

- Desarrollar un esquema de numeración para la identificación de MR/PR's. Por ejemplo podemos usar números de versión como 2.4-221098 e interpretar 2 como el numero de identificación del programa que recibe la petición de modificación, el 4 para indicar que es de tipo correctivo (usando 3 para adaptativo, el 2 para perfectivo etc.) y 221098 como la fecha en que se recibió.
- Desarrollar un esquema para categorizar y priorizar las MR/PR's. Es importante el análisis de las MR/PR ya que por ejemplo podría ocurrir que una petición que llega muy tarde necesite ser atendida la primera debido a la gravedad del error.
- Desarrollar procedimientos para analizar las tendencias. Esto nos permitirá "predecir" que clase de peticiones nos harán en el futuro.
- Determinar los procedimientos que debe ejecutar un operador para enviar un MR/PR. Por ejemplo obligar a que se entreguen por escrito siguiendo un formato determinado (donde se indique la clase de error, como se produce, cuando se produce, etc.)
- Determinar como se dará a los usuarios la realimentación inicial.
- Determinar como proporcionar a los usuarios jornadas de trabajo conjunto. Nos puede interesar hacer reuniones donde todo el mundo esté presente. Si esto no es posible intentar utilizar videoconferencias etc.
- Determinar como se introducen los datos según el estado de la base de datos.
- Determinar la realimentación posterior para los usuarios

### **8.1.2.3 Gestión de la configuración**

El mantenedor debería (ver norma ISO/IEC 12207 punto 5.5.1.3) implementar (o establecer una interfaz con) el Proceso de Gestión de Configuración para gestionar las modificaciones del sistema existente.

### **8.1.3 Controles**

Las revisiones conjuntas (ISO/IEC 12207 punto 6.6) se deberían usar para controlar las salidas de la Actividad de Implementación del Proceso.

### **8.1.4 Soporte**

La actividad de Implementación del Proceso usa los siguientes procesos de apoyo y del ciclo de vida de la organización (procesos de ISO/IEC 12207):

- Proceso de Documentación.
- Proceso de Gestión de la Configuración.
- Proceso de Aseguramiento de la Calidad.
- Proceso de Revisión Conjunta
- Proceso de Gestión
- Proceso de Infraestructura.
- Proceso de Entrenamiento.

### **8.1.5 Salidas**

Las salidas de ésta actividad son:

- El Plan de Mantenimiento
- Procedimientos para el mantenimiento
- Procedimientos para la resolución de problemas
- Planes para la realimentación del usuario.
- El Plan de Transición.
- Plan para la Gestión de la Configuración

Todas las actividades deberían estar bajo la gestión de la configuración.

## 8.2 Análisis de modificaciones y problemas

El análisis de los informes enviados es una actividad crítica ya que debemos entender el problema, desarrollar una solución y obtener la aprobación para poder desarrollarla. Durante la Actividad de Análisis de Modificaciones y Problemas el encargado de llevar a cabo el mantenimiento:

- Analiza los informes de problemas y propuestas de modificación (MR/PR), es decir intentamos comprender la raíz del problema.
- Replica o verifica el problema. Si el usuario no indica como se produce el error deberemos intentar producirlo por nosotros mismos y ver qué condiciones afectan al programa. Puede haber condiciones que el usuario no sepa que existan o simplemente no haya indicado.
- Desarrolla opciones para implementar la modificación. Siempre hay más de una forma de resolver los problemas. Deberemos documentarlas y encontrar sus ventajas e inconvenientes para que los directores de proyecto decidan la alternativa que vamos a implementar.
- Obtiene la aprobación para opción de modificación elegida.

La Entrada para la actividad de Análisis de modificaciones y problemas debería ser un informe de problema o petición de modificación validada, además de documentación sobre el Proyecto/Sistema y la documentación de requerimientos.

### 8.2.1 Entradas

Las entradas para la actividad de Análisis de Modificaciones y Problemas deberían ser:

- MR/PR. Lo normal es que nos haga falta el informe del problema que nos han dado.
- Línea a seguir. Deberemos tener una forma de comportamiento marcada por la empresa para tratar con el cliente.
- Repositorio de software. ¿Que conjunto de software tenemos en la empresa?.
- Documentación del sistema. Dentro de la cual se incluye:
  - Información del estado de la configuración. Debemos saber en qué estado nos encontramos.
  - Requerimientos funcionales. ¿Qué funciones da el software que tenemos?

- Requerimientos de interfaz. Algunos disminuidos físicos pueden necesitar interfaces especiales con tipos de letras grande.
- Datos de la planificación del proyecto
- Salidas de la Actividad de Implementación del Proceso.

## **8.2.2 Tareas**

Antes de modificar el sistema, el mantenedor debería analizar la MR/PR para determinar su impacto en la organización, en el sistema existente y en los sistemas conectados con él; desarrollar y documentar las posibles soluciones recomendadas, así como obtener la aprobación para implementar la solución deseada.

### **8.2.2.1 Análisis de las MR/PR's**

El encargado del mantenimiento debería analizar (ver ISO/IEC 12207 punto 5.5.2.1) el informe de problema o petición de modificación para ver su impacto en la organización, en el sistema existente y en los sistemas conectados con él para ver:

- Tipo: Por ejemplo modificación correctiva, preventiva, adaptativa a un nuevo entorno o petición de mejora.
- Alcance: Por ejemplo, tamaño de la modificación, coste en tiempo y dinero.
- Criticidad: Por ejemplo impacto en el rendimiento o seguridad.

Para asegurar que la MR/PR es factible el encargado del mantenimiento deberá seguir éstos pasos:

- Determinar si tenemos un personal adecuado para implementar el cambio propuesto.
- Determinar si el programa está bien presupuestado para implementar el cambio propuesto.
- Determinar si disponemos de los recursos suficientes y si esta modificación afectará a proyectos en curso o futuros.
- Determinar aspectos operacionales que nos afecten. Por ejemplo, cambios que podemos necesitar hacer en los requerimientos de interfaz, vida útil esperada del sistema, prioridades operacionales, seguridad física y lógica, etc.

- Determinar implicaciones en la seguridad física y lógica. Algunas peticiones de cambio podrían dar lugar a agujeros de seguridad.
- Determinar costes a largo y corto plazo
- Determinar el valor del beneficio si hacemos la modificación. No siempre tendremos en cuenta la rentabilidad, el no corregir errores debido a altos costes puede afectar a nuestra fama.
- Determinar el impacto en las planificaciones existentes. Si estamos llevando a cabo más de un proyecto puede que haya que quitarle recursos al otro.
- Determinar el nivel de pruebas y evaluaciones requerido. Algunos sistemas necesitan más pruebas que otros (no es igual el software de un avión que el de un programa de contabilidad).
- Determinar el coste de gestión de la implementación del cambio.

Debemos hacer notar que algunos de éstos cálculos pueden no ser necesarios en el caso de que recibamos un informe de problema, ya que están más referidos al impacto de cambios en el sistema.

### **8.2.2.2 Verificación**

Es muy recomendable que el encargado de llevar a cabo el mantenimiento (ver ISO/IEC 12207 punto 5.5.2.2) intente replicar o verificar el problema. Una vez que se recibe la MR se debería crear un registro en la base de datos de historiales de MR. Éste registro está diseñado para contener la información generada desde que se recibe hasta que se resuelve (es decir se implementa la MR o se cancela)

Para asegurar que los informes de problemas son válidos, el encargado debería hacer éstas tareas para la replicación o verificación:

- Desarrollar una estrategia de pruebas para verificar el problema.
- Obtener la versión afectada.
- Instalarla.
- Ejecutar pruebas para verificar el problema, preferiblemente con una copia de los datos afectados.
- Documentar los resultados de las pruebas

Si el problema no se puede replicar, por ejemplo porque los

datos son confidenciales se deberían comprobar otras cuestiones como reglas de la organización, políticas o documentación. La tarea de verificación no es necesaria en el mantenimiento adaptativo o perfectivo.

### **8.2.2.3 Opciones**

Basándose en el análisis, el mantenedor (ISO/IEC 12207 punto 5.5.2.3) debería desarrollar las distintas opciones para la implementación de la modificación:

- Asignando una prioridad a la MR/PR. Ésta prioridad puede depender de la política de la empresa.
- Definir los requerimientos de la compañía.
- Estimación del tamaño y magnitud de la modificación.
- Desarrollar al menos tres opciones para implementar la modificación.
- Determinar los impactos que éstas opciones tendrán en el hardware del sistema.
- Hacer un análisis de los riesgos que cada opción tiene.

### **8.2.2.4 Documentación**

Se debe documentar (ISO/IEC 12207 punto 5.5.2.4) el informe de problema o propuesta de modificación, los resultados del análisis y las opciones de implementación, llevando a cabo los siguientes pasos:

- Verificar que los análisis apropiados y la documentación del proyecto están actualizados. Si no existen desarrollar la documentación.
- Revisar la estrategia de pruebas propuesta y planificación para intentar obtener una exactitud mayor.
- Revisión de la estimación de recursos para una mayor exactitud.
- Actualizar el estado de la base de datos.
- Incluir una Recomendación para indicar si la MR/PR debería aprobarse o no. Al documentar el informe se deben dar a los directores de proyecto la recomendación personal sobre la MR/PR, lo que les permitirá tener una segunda opinión a la hora de decidir.

### **8.2.2.5 Aprobación**

Antes de modificar el sistema el mantenedor debería (ver ISO/IEC 12207 punto 5.5.2.5) obtener la aprobación de la opción de modificación elegida tal como se especifica en el contrato.

La aprobación debería obtenerse cuando el mantenimiento se lleva a cabo o cuando no se hace uso de los acuerdos para iniciar el mantenimiento (por ejemplo por algún error grave). Podemos obtener esta aprobación siguiendo éstos pasos:

- Presentar los resultados del análisis para su aprobación por parte de los grupos CM.
- Participar en las discusiones acerca de la modificación.
- Una vez aprobada la modificación actualizar el estado de la petición de modificación.
- Una vez aprobada actualizar también los requerimientos (en caso de que la petición sea una mejora)

### **8.2.3 Controles**

El control se mantiene por medio de revisiones conjuntas (ISO/IEC 12207 punto 6.6).

Al final de ésta actividad deberíamos llevar a cabo un análisis de riesgos. Usando las salidas de la actividad de Análisis de Problemas y Modificaciones dentro del Proceso de Mantenimiento se deberían revisar las estimaciones preliminares de recursos y tomar una decisión junto al usuario sobre si procedemos a ejecutar la actividad de Implementación de la Modificación.

### **8.2.4 Soporte**

La Actividad de Análisis de Modificaciones y Problemas usa los siguientes procesos del ciclo de vida de ISO/IEC 12207:

- Proceso de Documentación
- Proceso de Aseguramiento de la Calidad
- Proceso de Información acerca de los Problemas

### **8.2.5 Salidas**

Las salidas de ésta actividad son:

- **Análisis de Impactos:** La actividad nos devuelve como resultado un informe donde se detalla el efecto que provocarían los diferentes cambios en forma de gasto de tiempo/dinero
- **Opción Recomendada.** Basándonos en el informe anterior y tras evaluar los impactos elegiremos la opción a seguir.
- **Modificación Aprobada.** Documento por escrito en el que se da la respuesta afirmativa a la modificación.
- **Documentación Actualizada.** Registraremos en la base de datos de historia el proceso que hemos llevado a cabo. Ésto nos permitirá en el futuro hacer análisis estadístico sobre los datos y “predecir” que acciones deberemos ejecutar.

El análisis de impactos debería incluir los siguientes puntos:

- **Declaración del problema o nuevo requerimiento.** Debemos dar una especificación clara y sin ambigüedades.
- **Evaluación del problema o requerimiento.** ¿El problema es muy grave?
- **Clasificación del tipo de mantenimiento requerido.** Las consabidas correcciones adaptativas, perfectivas, correctivas y preventivas.
- **Prioridad Inicial.** ¿Cuál es nuestra evaluación actual de los riesgos?
- **Datos de Verificación** (para modificaciones correctivas)
- **Estimación Inicial de los recursos requeridos para modificar el sistema existente.**

La documentación actualizada debería incluir:

- **Una Estrategia de Pruebas.** Dependiente de la política de la empresa.
- **Documentación Actualizada sobre las Pruebas,** incluyendo el plan de pruebas, procedimientos para las pruebas e informes sobre las pruebas.
- **Carpetas con información sobre el desarrollo de software**
- **Requerimientos Actualizados**



## 8.3 Implementación de la Modificación

Durante la Actividad de Implementación de la Modificación, el mantenedor desarrolla y prueba la modificación del producto software.

### 8.3.1 Entradas

Las entradas a la actividad de Implementación de la Modificación son:

- Línea base a seguir
- La MR/PR aprobada
- La Documentación de la Modificación Aprobada

La línea base debería incluir:

- Definiciones sobre la Arquitectura del Sistema
- El Registro de la Petición de Modificación
- Código fuente

La Documentación de la Modificación Aprobada debería incluir:

- El informe sobre el Análisis de Impacto
- Salidas de la Actividad de Análisis de Modificaciones y Problemas

### 8.3.2 Tareas

El mantenedor realiza un análisis, y después lleva a cabo el Proceso de Desarrollo de ISO/IEC 12207 para efectuar la modificación.

#### 8.3.2.1 Análisis

Una vez aprobada la propuesta de modificación o el informe sobre problema el mantenedor debería dirigir el análisis y determinar que documentación, unidades de software y versiones deben ser modificadas. Todo esto debería ser documentado (ISO/IEC 12207 punto 5.5.3.1).

Los resultados de éste análisis adicional debería documentarse en las Carpetas sobre el Desarrollo de Software. Éste esfuerzo supondrá:

- Identificar los elementos a modificar en el sistema existente

- Identificar los elementos del interfaz afectados por la modificación
- Identificar la documentación a actualizar
- Actualizar las Carpetas sobre el Desarrollo de Software

### **8.3.2.2 Proceso de Desarrollo**

El mantenedor debería (ver ISO/IEC 12207 punto 5.5.3.2) entrar en el Proceso de Desarrollo (ISO/IEC 12207 punto 5.3) para implementar las modificaciones. Los requerimientos del Proceso de Desarrollo debería (ISO/IEC 12207 punto 5.5.3.2) complementarse con:

1. Se deberían documentar y definir criterios de pruebas y evaluación para la comprobación y evaluación de las partes modificadas y no modificadas (unidades software, componentes y elementos de configuración).
2. Se debe garantizar la implementación correcta y completa de los requerimientos nuevos y modificados (ISO/IEC 12207 punto 5.5.3.2). También se debería (ISO/IEC 12207) asegurar que los requerimientos originales, sin modificar no están afectados. Los resultados de las pruebas deberían (ISO/IEC 12207 punto 5.5.3.2) documentarse

Las actividades del Proceso de Desarrollo deberían personalizarse a las necesidades del esfuerzo de modificación.

### **8.3.3 Controles**

La Implementación del Control de Modificaciones debería incluir revisiones conjuntas (ver ISO/IEC 12207 punto 6.6)

### **8.3.4 Soporte**

La Actividad de Implementación de la Modificación utiliza los siguientes procesos del ciclo de vida de ISO/IEC 12207:

- Proceso de Documentación
- Proceso de Aseguramiento de la Calidad
- Proceso de Revisión Conjunta

### **8.3.5 Salidas**

Las salidas de ésta actividad deberían incluir:

- Planes y Procedimientos para las Pruebas Actualizados
- Documentación Actualizada
- Código Fuente modificado
- Informe de Pruebas
- Métricas

La documentación actualizada debería incluir

- Registros de Modificaciones Actualizados
- Informe Detallado sobre el Análisis
- Requerimientos Actualizados
- Planes, Informes y Procedimientos sobre las Pruebas actualizados
- Material de Entrenamiento actualizado

## **8.4 Aceptación/Revisión del Mantenimiento**

Ésta actividad asegura que las modificaciones al sistema se han hecho de forma correcta y de acuerdo a los estándares aprobados dentro del uso de una metodología correcta.

### **8.4.1 Entradas**

Las entradas a la actividad de Aceptación/Revisión del mantenimiento son:

- El Software Modificado
- Resultados de las Pruebas de la Modificación

### **8.4.2 Tareas**

Las revisiones se dirigen de forma que aseguremos que las modificaciones son correctas y que se obtenga un final satisfactorio de la modificación

### **8.4.2.1 Revisiones**

El mantenedor debería (ISO/IEC 12207 punto 5.5.4.1) dirigir la revisión con la organización que autoriza la modificación para determinar la integridad del sistema modificado

Se deben llevar a cabo los siguientes pasos/tareas:

- Seguir la pista al informe de problema o propuesta de modificaciones desde los requerimientos hasta su transformación en código
- Verificar que el código es comprobable
- Verificar que se cumple con los estándares de codificación
- Verificar que solo se han modificado los componentes software necesarios
- Verificar que los nuevos componentes software se integran de forma correcta
  - Comprobar la documentación para asegurar que está actualizada
  - Ejecutar pruebas
  - Desarrollar un informe sobre las pruebas

### **8.4.2.2 Aprobación**

El mantenedor debería (ISO/IEC 12207 punto 5.5.4.2) obtener la aprobación para completar de forma satisfactoria la modificación tal como se especifica en el contrato.

Si el mantenimiento se implementó sin un acuerdo debería obtenerse de todas formas una aprobación. Se deberían llevar a cabos los siguientes pasos/tareas:

- Obtener la aprobación por medio de los procesos de soporte del ciclo de vida para el aseguramiento de la calidad (ISO/IEC 12207)
- Verificar que se ha seguido el proceso
- Dirigir auditorías de configuración física y funcional

### **8.4.3 Controles**

El control se ejercita por medio de revisiones conjuntas (ver ISO/IEC 12207 punto 6.6)

#### **8.4.4 Soporte**

La actividad de Aceptación/Revisión del Mantenimiento usa los siguiente procesos del ciclo de vida para el soporte:

- Proceso de Aseguramiento de la Calidad
- Proceso de Verificación
- Proceso de Validación
- Proceso de Revisión Conjunta
- Proceso de Auditoría

#### **8.4.5 Salidas**

Las salidas de ésta actividad son:

- Nueva línea base, incorporando las modificaciones aceptadas
- Modificaciones rechazadas
- Informe de aceptación
- Informes de revisión y auditoría
- Informe de pruebas de cualificación del software

### **8.5 Migración**

Durante la vida de un sistema, puede que haya que modificarlo para ejecutarlo en entornos diferentes. Para migrar un sistema a un nuevo entorno, el mantenedor necesita determinar las acciones necesarias para conseguir la migración y a partir de ahí desarrollar y documentar los pasos necesarios para efectuar la migración

#### **8.5.1 Entradas**

Las entradas a la actividad de Migración son:

- El Antiguo Entorno

- El Nuevo Entorno
- La Antigua Línea Base
- La Nueva Línea Base

## **8.5.2 Tareas**

El mantenedor efectúa la migración de forma que cumpla las normas ISO/IEC 12207, desarrollando un plan de migración, notificando a los usuarios la migración, proporcionando entrenamiento, avisando del término de la migración, calculando el impacto del nuevo entorno y archivando datos

### **8.5.2.1 Migración**

Si un sistema o producto software (incluyendo sus datos, migra a un nuevo entorno operativo, se debería asegurar (ISO/IEC 12207 punto 5.5.5.1) que cualquier dato o producto software producido o modificado durante la migración cumple la norma ISO/IEC 12207.

Se deberían llevar a cabo los siguientes pasos o tareas:

- Identificar todos los productos software o datos que se van a añadir o modificar
- Verificar que las tareas cumplen la norma ISO/IEC 12207

### **8.5.2.2 Plan de migración**

Para que se pueda controlar de forma adecuada la migración de un sistema, deberíamos crear (ISO/IEC 12207 punto 5.5.5.2) un plan de migración, además de documentarlo y ejecutarlo. Las actividades de planificación deberían (ISO/IEC 12207 punto 5.5.5.2) incluir:

- Análisis de requerimientos y definición de la migración
- Desarrollo de herramientas de ayuda a la migración
- Conversión de datos y productos software
- Ejecución de la migración
- Verificación de la migración
- Soporte para el antiguo entorno

El desarrollo del Plan de Migración debería incluir las entradas que puedan proporcionarnos los usuarios. Como parte de ésta

tarea, el mantenedor debería ejecutar los siguientes pasos:

- Analizar los requerimientos de la migración
- Determinar el impacto de la migración del producto software
- Establecer una planificación para efectuar la migración
- Identificar los requerimientos de los conjuntos de datos para su revisión posterior
- Definir y documentar el esfuerzo de migración
- Determinar y mitigar los riesgos
- Identificar las herramientas de migración necesarias
- Desarrollar y/o comprar las herramientas de migración
- Descomponer los productos y datos software de forma incremental para su conversión
- Convertir los productos y datos software
- Migrar los productos y datos software al nuevo entorno
- Ejecutar las operaciones paralelas
- Verificar la migración por medio de pruebas
- Proporcionar soporte para el antiguo entorno

### **8.5.2.3 Notificación del intento**

Una vez que el encargado del mantenimiento haya completado la planificación de la migración, los usuarios deberían (ISO/IEC 12207 punto 5.5.5.3 recibir la notificación de los planes y actividades de migración. Dentro de éstas notificaciones debería haber:

- Explicación de por qué ya no da soporte al antiguo entorno
- Descripción del nuevo entorno junto a la fecha de disponibilidad
- Descripción de otras opciones de soporte disponibles, si las hay, una vez que hayamos abandonado el antiguo entorno

El mantenedor también debería proporcionar a los usuarios el plan, los procedimientos, y la planificación. Como parte de ésta tarea, el mantenedor debería llevar a cabo los siguientes pasos:

- Identificar los puestos afectados por la migración

- Obtener la realimentación del puesto afectado
- Identificar aspectos específicos del puesto
- Divulgar la planificación

#### **8.5.2.4 Implementación de las operaciones y entrenamiento**

Las operaciones paralelas del antiguo y nuevo entorno deberían dirigirse de forma que la transición del viejo al nuevo entorno sea suave (ISO/IEC 12207 punto 5.5.5.4). Durante éste período, deberíamos proporcionar de acuerdo con el contrato (ISO/IEC 12207 punto 5.5.5.4) el entrenamiento necesario.

Como parte de ésta tarea, el mantenedor debe realizar los siguientes pasos:

- Conservar un puesto en su configuración original
- Instalar el equipo
- Instalar el software
- Ejecutar algunas pruebas preliminares para asegurarnos de una correcta instalación del hardware y del software
- Ejecutar el software con una carga operativa en el entorno antiguo y en el nuevo
- Recoger datos de los productos nuevos y viejos
- Reducir y analizar los datos

El mantenedor, si quiere dar un correcto entrenamiento debería:

- Identificar los requisitos de entrenamiento
- Planificar los requisitos de entrenamiento
- Dirigir la revisión del entrenamiento
- Actualizar los planes de entrenamiento

#### **8.5.2.5 Notificación del final**

Una vez lleguemos al final de la migración planificada, se debería (ISO/IEC 12207 punto 5.5.5.5) enviar la notificación a todos los interesados. Toda la documentación asociada al antiguo entorno, así como los registros y código se deberían archivar (ISO/IEC 12207 punto



#### 5.5.5.5)

Como parte de ésta tarea, el mantenedor debería:

- Divulgar los cambios de la planificación de la migración
- Documentar los aspectos específicos del puesto y como se resolverán
- Archivar los datos y el software viejos
- Retirar el antiguo equipo

### **8.5.2.6 Revisión post-operación**

Este proceso se debe realizar para calcular el impacto de cambios a un nuevo entorno. Los resultados de la revisión deberían (ISO/IEC 12207 punto 5.5.5.6) enviarse a las autoridades apropiadas para su información, guía y actuación.

Como parte de éste riesgo sería aconsejable que el mantenedor:

- Revisar los resultados al operar con los dos entornos a la vez
- Identificar las áreas con un riesgo potencial
- Identificar aspectos específicos del puesto
- Documentar las lecciones aprendidas
- Generar y anticipar un informe sobre el Análisis del Impacto

### **8.5.2.7 Archivado de datos**

Los datos usados por o asociados con el antiguo entorno se deberían conservar accesibles de acuerdo con los requisitos del contrato para su protección o auditoría.

Esta tarea se descompondría en las siguientes subtareas:

- Almacenar los datos y el software viejos.
- Hacer copias de los datos y el software viejos.
- Almacenar las copias en un lugar seguro.

### **8.5.3 Controles**

El control se lleva a cabo por medio de revisiones conjuntas (ISO/IEC 12207 punto 6.6)

### **8.5.4 Soporte**

La actividad de Migración usa los siguientes procesos del ciclo de vida de ISO/IEC 12207 relativos a la organización y al soporte:

- Proceso de Documentación
- Proceso de Gestión de la Configuración
- Proceso de Aseguramiento de la Calidad
- Proceso de Verificación
- Proceso de Validación
- Proceso de Revisión Conjunta
- Proceso de Auditoría
- Proceso de Informe sobre problemas
- Proceso de Entrenamiento

### **8.5.5 Salidas**

Las salidas de ésta actividad son:

- Plan de Migración
- Herramientas de Migración
- Notificación de Intentos
- Producto Software Migrado
- Notificación de Finalización
- Datos archivados

## **8.6 Retiro del software**

Una vez que el producto ha alcanzado el final de su vida útil debe retirarse. Se debería hacer un análisis para ayudar en la toma de la decisión de retiro de un producto software. El análisis a menudo está basado en aspectos económicos y debería incluirse en el Plan de Retiro. Deberíamos hacer un análisis para ver si es efectivo en cuanto al costo el:

- Conservar software obsoleto
- Pasar a una nueva tecnología desarrollando un nuevo producto software
- Desarrollar un nuevo producto software para conseguir modularidad
- Desarrollar un nuevo producto software para facilitar el mantenimiento
- Desarrollar un nuevo producto software para lograr la estandarización
- Desarrollar un nuevo producto software para alcanzar la independencia del fabricante

El producto software podría reemplazarse por un nuevo producto software pero no siempre. Para retirar un producto software, el mantenedor debería determinar las acciones necesarias para conseguir el retiro y entonces desarrollar y documentar los pasos necesarios para efectuar el retiro. Deberíamos tener en cuenta los datos almacenados por el producto software retirado

### **8.6.1 Entradas**

Las entradas a la actividad de retiro son:

- El producto software a retirar
- El nuevo producto software
- El antiguo entorno

### **8.6.2 Tareas**

El mantenedor efectúa el retiro de forma que cumpla el estándar ISO/IEC 12207, desarrollando un plan de retiro, notificando a los usuarios dicho retiro, notificando la finalización de la actividad de retiro y archivando los datos

#### **8.6.2.1 Plan de retiro**

Se debería (ISO/IEC 12207 punto 5.5.6.1) desarrollar y documentar un plan de retiro para eliminar el soporte por parte de las organizaciones que operan con el sistema y lo mantienen. Las actividades de planificación deberían (ISO/IEC 12207 punto 5.5.6.1)

incluir a los usuarios. El plan debería (ISO/IEC 12207 punto 5.5.6.1) tener en cuenta los aspectos siguientes:

- Cese del soporte total o parcial tras cierto período de tiempo
- Archivado del software y su documentación asociada
- Responsabilidad de futuros aspectos de soporte residuales
- Transición al nuevo producto software (si hubo un antiguo)
- Accesibilidad a las copias archivadas de los datos

Como parte de ésta tarea, el mantenedor debería:

- Analizar los requerimientos de retiro
- Determinar el impacto del retiro del producto software
- Identificar claramente el producto software a reemplazar, si lo hay
- Establecer una planificación para el retiro del producto software
- Identificar los responsables del soporte residual futuro
- Definir y documentar el esfuerzo de retiro

### **8.6.2.2 Notificación del intento**

Se debería (ISO/IEC 12207 punto 5.5.6.2) notificar a los usuarios los planes de retiro y actividades. Ésta notificación debería incluir:

- Descripción del reemplazamiento o actualización con su fecha de disponibilidad
- Declaración de por qué el producto software no seguirá recibiendo soporte
- Descripción de otras opciones de soporte disponibles, una vez que el producto software haya sido retirado

En éste caso deberemos:

- Identificar todos los puestos afectados
- Identificar los aspectos específicos del puesto
- Divulgar la planificación

- Procesar la realimentación procedente del puesto

### **8.6.2.3 Implementar las operaciones paralelas y de entrenamiento**

Las operaciones paralelas de retiro del viejo software e implantación del nuevo deberían conducirse de forma que haya una transición suave al nuevo sistema. Durante éste período el usuario debería ser entrenado de la forma en que se especifique en el contrato.

Aquí tendremos que:

- Conservar un puesto en su configuración original
- Instalar el equipo
- Instalar el software
- Ejecutar algunas pruebas preliminares para asegurarnos de una correcta instalación del hardware y del software
- Ejecutar el software con una carga operativa en el entorno antiguo y en el nuevo
- Recoger datos de los productos nuevos y viejos
- Reducir y analizar los datos

### **8.6.2.4 Notificación de finalización**

Una vez lleguemos al final de la migración planificada, se debería (ISO/IEC 12207 punto 5.5.5.5) enviar la notificación a todos los interesados. Toda la documentación asociada al antiguo entorno, así como los registros y código se deberían archivar (ISO/IEC 12207 punto 5.5.5.5)

Como parte de ésta tarea, el mantenedor debería:

- Divulgar los cambios de la planificación de la migración
- Documentar los aspectos específicos del puesto y como se resolverán
- Archivar los datos y el software viejos
- Retirar el antiguo equipo

### **8.6.2.5 Archivado de datos**

Los datos usados por o asociados con el antiguo entorno se deberían conservar accesibles de acuerdo con los requisitos del contrato para su protección o auditoría.

Esta tarea se descompondría en las siguientes subtarear:

- Almacenar los datos y el software viejos.
- Hacer copias de los datos y el software viejos.
- Almacenar las copias en un lugar seguro.

### **8.6.3 Controles**

El control se lleva a cabo por medio de revisiones conjuntas (ISO/IEC 12207 punto 6.6)

### **8.6.4 Soporte**

La actividad de Retiro del Software usa los siguientes procesos del ciclo de vida de ISO/IEC 12207 relativos a la organización y al soporte:

- Proceso de Documentación
- Proceso de Gestión de la Configuración
- Proceso de Aseguramiento de la Calidad
- Proceso de Revisión Conjunta
- Proceso de Auditoría
- Proceso de Entrenamiento

### **8.6.5 Salidas**

Las salidas de ésta actividad son:

- Plan de Retiro
- Notificación de Intento
- Resultados del Retiro
- Personas entrenadas
- Producto Software Retirado

- Notificación de finalización
- Línea base del producto retirado archivada.

## BIBLIOGRAFÍA

- Norma ISO 14764 sobre mantenimiento de software. Diversos autores
- Mantenimiento del Software, conceptos, métodos, herramientas y outsourcing. Diversos autores.
- Practical Software Maintenance. Thomas M. Pigoski
- Ingeniería del Software. Robert L. Pressman.
- IEEE International Software Maintenance Workshop. Diversos autores