

UNIVERSIDAD NACIONAL AUTONOMA DE MÉXICO
FACULTAD DE ESTUDIOS SUPERIORES "ACATLAN"

**PROTOCOLO PARA INTEGRAR UNA CARPETA DE CALIDAD PARA LA
ADMINISTRACIÓN DE PROYECTOS DE MANTENIMIENTO DE
PROGRAMAS DE SOFTWARE DE GESTIÓN.**

TESINA

**TESIS CON
FALLA DE ORIGEN**

QUE PARA OBTENER EL TÍTULO DE:

LICENCIADO EN MATEMATICAS APLICADAS Y COMPUTACIÓN.

PRESENTA:

MARIA SUSANA PÉREZ ROSALES

ASESOR: PROFESORA TITULAR "A", T.C.,DEFINITIVA ING. BEATRIZ CLAVEL DÍAZ

MÉXICO, D.F., NOVIEMBRE DEL 2005

m 340769



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



Dedico este trabajo a
A mis padres, Victoria y Miguel por su cariño y amor.
A Javier mi esposo por ser el impulso de mi vida.
Y a mi profesora Clavel, a quien respeto y admiro.

Objetivo:

Establecer el protocolo para la integración de una carpeta de calidad para la administración de proyectos de mantenimiento de software de gestión con fundamento en los criterios de calidad que señala la norma ISO 9000.

ÍNDICE

Introducción.....	1
Capítulo 1. MANTENIMIENTO DE SOFTWARE DE GESTIÓN	
1. Marco Teórico conceptual para el mantenimiento de Software.....	3
1.1 Características de software.....	4
1.2 Fallos y Deterioros y envejecimiento de software.....	4
1.3 Clasificación del software de acuerdo a su aplicación.....	6
1.4 Paradigmas de la Ingeniería de software.....	7
1.4.1 Ciclo de Vida Clásico.....	10
1.5 Mantenimiento de software.....	11
1.5.1 Actividades y Tipos.....	12
1.6 Problemas asociados al mantenimiento de software.....	14
1.7 Facilidad de Mantenimiento.....	14
1.8 Tareas de Mantenimiento.....	15
1.8.1 Registro de Información	15
1.8.2 Documentación de los cambios del programa.....	16
1.9 Gestión de configuraciones.....	17
1.9.1 Líneas Base.....	17
1.9.2 Elementos de configuración de software.....	18
Resumen.....	21
Capítulo 2. ADMINISTRACION DE PROYECTOS DE MANTENIMIENTO DE SOFTWARE DE GESTION.	
2.1 Tipos de Proyectos de Mantenimiento de Software.....	22
2.1.1 Características.....	23
2.2 Ciclo de Vida de los Proyectos.....	24
2.3 Diagramas de Gantt.....	32
Resumen.....	34

Capítulo 3. **PROTOCOLO PARA INTEGRAR LA CARPETA DE CONTROL DE PROYECTOS DE MANTENIMIENTO DE SOFTWARE DE GESTION.**

3.1	Normas para la Gestión de Calidad.....	36
3.1.1	Concepto General de la Calidad.....	37
3.2	¿Qué es ISO?.....	37
3.2.1	Normas ISO9000.....	38
3.2.1.1	¿Quién Desarrollo la serie ISO 9000?.....	38
3.2.1.2	¿En que consiste la serie ISO 9000?.....	38
3.2.1.3	Calidad en productos y servicios.....	39
3.2.1.4	Cláusulas de la Norma ISO 9001.....	40
3.3	Protocolo para la integración de la carpeta de calidad.....	42
	Conclusión.....	69
	Anexo A.....	71
	Anexo B.....	73
	Referencias Bibliográficas.....	78
	Referencias Electrónicas.....	79

INTRODUCCION

Este trabajo esta dirigido a personas que estén interesadas en conocer un protocolo que sirva de guía para generar la documentación oportuna obtenida durante la administración de un proyecto de mantenimiento de programas de software de gestión a través de los cuales puede brindarse retroalimentación, detectar desviaciones de los objetivos establecidos, así como tener un mejor control sobre el desarrollo del proyecto y por supuesto conservar una memoria perenne.

El objetivo general de esta tesina: **establecer un protocolo que permita integrar un portafolio o carpeta con la documentación obtenida en los Proyectos de Mantenimiento de Programas de Software de Gestión (PMPSG)**, surge debido a que en la administración de dichos proyectos no existen procedimientos que normen o regulen la generación y registro de documentos ad hoc para el control del proyecto.

Un proceso documentado ayuda a controlar las actividades del proyecto influyendo de manera positiva en la calidad. La documentación no debe ser excesiva, ni tampoco paupérrima, por este motivo se eligió como referencia a la norma de calidad ISO 9001, la cual es el prototipo de conformidad más completo de la serie ISO 9000 y es reconocida mundialmente; esta señala los puntos más importantes que deben de documentarse en un sistema.

En este trabajo se refiere a la actividad del mantenimiento porque existe poca información acerca del tema de los proyectos de mantenimiento de software, además puede llegar a consumir hasta el 50 % del presupuesto de las empresas;

Cabe señalar que este estudio es total y estrictamente descriptivo desde la perspectiva del mantenimiento de sistemas y los aportes de la Ingeniería de Software. En consecuencia, metodológicamente se trata de una investigación documental en la que a partir del análisis de los principios, postulados y procedimientos propuestos por la norma de calidad vigente ISO 9000, se hace una recopilación conceptual racional e intuitiva de términos y procedimientos. Se analiza la papelería que puede obtenerse en cada fase del ciclo de vida de la administración de un Proyecto de Mantenimiento de Programas de Software de Gestión (PMPSG) para posteriormente contrastarlos con lo prescrito en la norma de calidad ISO 9001.

Para alcanzar el objetivo pretendido, se dividió el análisis en tres partes. Resultó indispensable incorporar un primer capítulo sobre la ingeniería de software con el propósito de plasmar los fundamentos conceptuales que dieron sustento a la actividad del mantenimiento, dado que esta actividad es, sino estrictamente igual, es similar al diseño del software, toda vez que la ingeniería de software compila principios, técnicas y procedimiento para el diseño y desarrollo de un software nuevo y el mantenimiento ocupa estos mismos principios para el mantenimiento de programas de software de gestión ya liberados al usuario.

El segundo capítulo extrapola el ciclo de vida de la ingeniería de software a los proyectos de mantenimiento, y señala los documentos que se pueden generar u obtener en cada una de sus fases.

En el tercer capítulo se citan los conceptos de calidad que sirven de criterio para conformar el protocolo, se expone el protocolo y con el desarrollo de un ejemplo se vislumbra el alcance que tiene el establecimiento de un protocolo para integrar una carpeta de calidad para los PMPSG. Considerando el alcance del protocolo, se proponen nuevas líneas de estudio, para posibles temas de tesis o tesinas.

Finalmente, cabe señalar que el protocolo propuesto no pretende ser una panacea para el control de la documentación de los Proyectos de Mantenimiento de Programas de Software de Gestión mencionados, tan sólo pretende brindar un camino para ayudar al responsable a coordinar un plan predeterminado. Estratégicamente, para darle seguimiento al proceso de mantenimiento de programas de software de gestión, subyace a esta propuesta un principio de ordenamiento cronológico de documentación de cada una de las fases de su ciclo de vida.

MANTENIMIENTO DE SOFTWARE DE GESTION

Objetivos específicos: Mostrar a la ingeniería de software como un elemento clave para facilitar el mantenimiento de programas. Exponer las características fundamentales del software para establecer la diferencia que existe entre el mantenimiento de software y el mantenimiento de hardware. Mencionar los diferentes tipos de mantenimiento existentes y sus características.

1. Marco Teórico Conceptual para el mantenimiento de software

La ingeniería del software es una disciplina centrada en el desarrollo de software de alta calidad para sistemas basados en computadora, consta de tres fases: definición, desarrollo y mantenimiento. La fase de definición empieza con la etapa de planificación, posteriormente se diseña, desarrolla y prueba. En la fase de desarrollo, se realiza una revisión de la configuración para asegurar que toda la documentación está disponible y es la adecuada para las tareas de mantenimiento a seguir, se define entonces, un esquema para las modificaciones del sistema y los errores. Finalmente, en la fase de mantenimiento del software, se corrigen errores, se hacen adaptaciones y se implementan mejoras.

En todos los casos el mantenimiento del software incluye la configuración entera (documentos desarrollados en las fases de planificación y desarrollo) y no sólo el código.

1.1 Características del software

Para poder comprender lo que es el software, es importante examinar las características del software que lo hacen diferente de otras cosas que los hombres pueden construir. Cuando se construye el hardware, el proceso creativo de análisis, diseño, construcción y prueba, se traduce en componentes físicos, no así en el caso del software puesto que *es un componente lógico en vez de físico del sistema*, y sus características son distintas a las del hardware.

El software es desarrollado, no manufacturado en el sentido clásico. El software es un elemento intangible que al igual que los conocimientos no se puede tocar.

1.2 Fallos, deterioros y envejecimiento del software

El software no se “deteriora” de manera física como en el caso del hardware.

La figura 1.1 describe la proporción de fallos como una función de tiempo, para el hardware. La relación llamada frecuentemente “curva de bañera”, indica que el hardware exhibe relativamente muchos fallos al principio de su vida. Estos fallos son atribuibles frecuentemente a defectos de diseño o de fabricación. Cuando se corrigen los defectos, caen los fallos hasta su nivel más bajo, donde permanecen estacionarios durante un cierto periodo. Sin embargo, al paso del tiempo, los fallos vuelven a presentarse conforme las componentes del hardware sufren los efectos acumulativos de suciedad, vibración, uso, temperaturas extremas y otras muchas cosas.

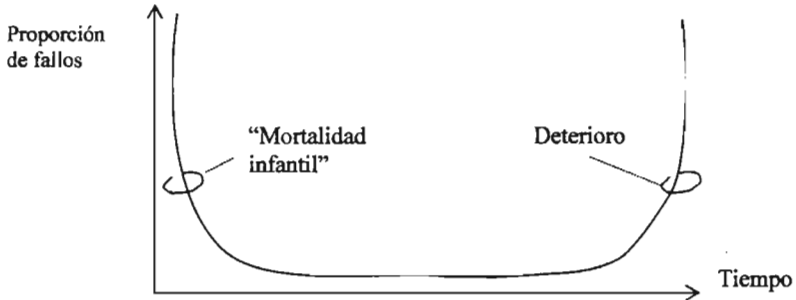


Fig. 1.1. Curva de fallos de hardware

Hablando estrictamente, el software no es susceptible de males del entorno físico, como los que hacen que el hardware se deteriore. Por tanto en teoría, la curva de fallos para el software tendría la forma mostrada en la figura 1.2.

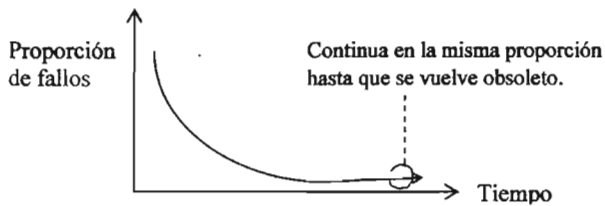
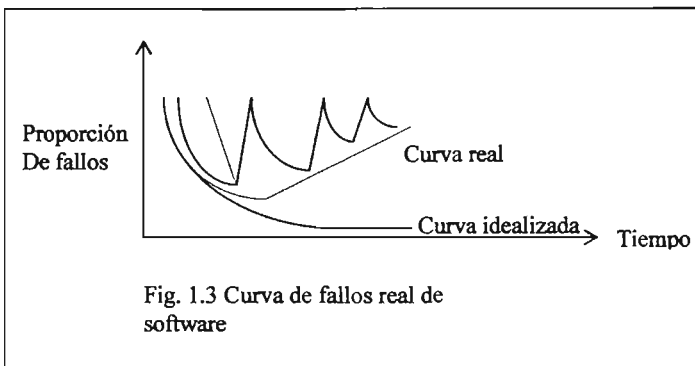


Fig. 1.2 Curva de fallos del software (idealizada)

Los defectos no descubiertos durante las pruebas de instalación, harán que falle durante las primeras etapas de la vida del programa. Sin embargo, una vez que los fallos se corrigen, suponiendo que no se introducen nuevos errores, la curva se aplanaría.

En realidad, esto no es así, puesto que durante su vida, el software, en general sufre cambios; de aquí que se requiera dar mantenimiento. Y conforme se hacen los cambios es probable que se introduzcan nuevos defectos, haciendo que la curva de fallos tenga picos como muestra la figura 1.3.



Antes de que la curva pueda volver al estado estacionario original, se solicita otro cambio, haciendo que se cree un nuevo pico y comúnmente el software se va deteriorando debido a los cambios, pero en el software no hay piezas de repuesto como las hay en el caso del deterioro del hardware. Cada fallo en el software indica un error en el diseño o en el proceso operacional en el que se tradujo el diseño en código máquina ejecutable.

1.3 Clasificación del software de acuerdo a aplicaciones

El software puede aplicarse en cualquier situación en la que se haya definido previamente un conjunto específico de pasos procedimentales¹. Es necesario señalar que estas clasificaciones no son excluyentes una de otras, es decir, puede existir combinaciones entre ellas.

¹ Procedimental: Método o manera de hacer las cosas (Diccionario de Sinónimos y Antónimos Larousse)

Las siguientes áreas del software indican la amplitud de las potenciales aplicaciones.

- Software de sistemas.
- Software de tiempo real.
- Software de Gestión.
- Software de Ingeniería y científico.
- Software Empotrado
- Software de Computadoras Personales (PC)

Para el propósito de este trabajo sólo se define el concepto de software de gestión, las demás definiciones se pueden encontrar en el anexo A (Pág. 69)

El software de gestión² es el que accede a una o más bases de datos en la que está contenida información comercial³. El procesamiento de información comercial constituye la mayor de las áreas de aplicación del software. Los “sistemas discretos” (Por ejemplo nominas, inventario, etc.) han evolucionado hacia el software de sistemas de información de gestión. Las aplicaciones en esta área reestructuran los datos existentes en orden para facilitar las operaciones comerciales o gestionar la toma de decisiones. Ejemplo de este software son los programas bancarios que accesan a grandes bases de datos y el procesamiento de transacciones en puntos de venta.

1.4 Paradigmas de la ingeniería de software

El software ha sufrido lo que muchas personas llaman “crisis del software” esto se refiere al conjunto de problemas encontrados en el desarrollo del software de computadoras.

² Nota: en este trabajo se utilizará la palabra *gestión y administración* de manera indistinta, puesto que tienen el mismo significado. El criterio de su utilización se basó en la manera en la que generalmente se ocupan en cada uno de los conceptos, por ejemplo dentro de la norma ISO 9000 se menciona como gestión de tal manera, por este motivo en el presente trabajo se hace referencia de la misma forma.

³ Este concepto se refiere a toda la información con la que se pueden hacer transacciones de compra y venta.

La ingeniería del software es una disciplina que combina métodos y herramientas para lograr una mejor coordinación, control y gestión.

Actualmente no existe una definición universal para la ingeniería de software; para algunos ingenieros de software, es sólo un nombre glorioso para la programación, sin embargo, otros tienen expectativas más altas sobre su concepto. En un libro de texto se puede encontrar una definición para este concepto, parecida a la siguiente:

“La ingeniería de software es el conjunto de métodos, herramientas y técnicas dirigidas para producir software con calidad”.

Otras personas afirman que se puede distinguir la ingeniería de software de programar por su naturaleza industrial, cómo la definición que a continuación se expone:

“El desarrollo de grandes sistemas dirigidos a trabajar en ambientes de producción, durante un largo periodo de tiempo, utilizado por muchas personas y sufriendo muchos cambios donde el desarrollo incluye la administración, el mantenimiento, la validación y su documentación...”

Otra definición propuesta para este concepto la declara Fritz Bauer⁴ (1969) en la primera conferencia celebrada para analizar la “crisis del software” y es la explicación que se toma en cuenta para este trabajo:

“El establecimiento y uso de principios de ingeniería robustos, orientados a obtener económicamente software que sea fiable y funcione eficientemente sobre máquinas reales”.

⁴ Naur, P., y B. Randell (eds.) Software Engineering: A report on a Conference sponsored by the NATO Science Committee, NATO, 1969.

De aquí se desprende que la ingeniería de software abarca un conjunto de tres elementos claves: Métodos, herramientas y procedimientos.

Los *métodos* representan el “cómo” construir técnicamente el software. Estos métodos abarcan un amplio espectro de tareas, que incluyen:

1. Planificación y estimación de proyectos
2. Análisis de los requerimientos del sistema y del software
3. Diseño de estructura de datos, arquitectura de los programas y procedimientos algorítmicos
4. Codificación
5. Prueba
6. **Mantenimiento**

Las *herramientas* suministran un soporte automático para los métodos. Cuando se integran las herramientas de forma que la información creada por una herramienta pueda ser usada por otra, se establece un sistema para el soporte del desarrollo del software, llamado ingeniería de software asistido por computadora.

Los *procedimientos* unen a las herramientas y métodos, facilitando un desarrollo racional y oportuno del software de computadora. Definen la secuencia en que se aplican los métodos, las entregas (documentos, informes, formas, etc.) que se requieren, los controles que ayudan a asegurar la calidad y coordinar los cambios y las guías.

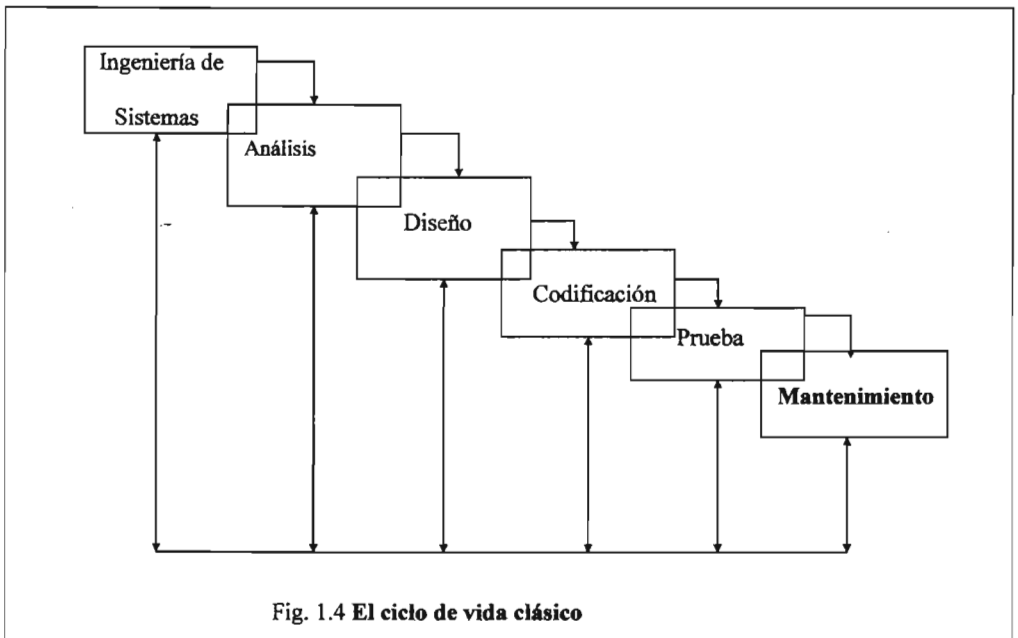
La ingeniería de software está compuesta de pasos o paradigmas que abarcan los métodos, herramientas y procedimientos. Cada paso se elige basándose en la naturaleza del proyecto, de la aplicación, de los métodos y herramientas a usar y los controles y entregas requeridos.

1.4.1 El ciclo de vida clásico en la ingeniería de software

Dentro de la ingeniería de software existen varios tipos de modelos de ciclo de vida, entre los cuales se encuentra el modelo en espiral, o el desarrollo incremental (los cuales se describen en el anexo B).

La figura 1.4 ilustra otro modelo a veces llamado el *modelo en cascada*, el paradigma de ciclo de vida ofrece un modelo sistemático, secuencial, del desarrollo del software que comienza en el nivel del sistema y progresa a través del análisis, el diseño, codificación, prueba y mantenimiento. Abarca las siguientes actividades:

- **Ingeniería y análisis del sistema.** Aquí se establecen los requerimientos del sistema, con una pequeña cantidad de análisis y diseño.
- **Análisis de los requerimientos del software.** Aquí se documentan y se revisan con el cliente los requerimientos tanto del sistema como del software.
- **Diseño.** Aquí se traducen los requerimientos en una representación del software y se documenta de manera detallada.



- **Prueba.** Enfocada en la lógica interna del software, probando cada una de las sentencias y las funciones externas.
- <<**Mantenimiento**>> El software sufrirá indudablemente cambios después de que sea entregado al cliente. Los cambios ocurrirán porque se han encontrado errores, debido a que el software debe adaptarse por cambios del entorno externo, o debido a que el cliente requiere aumentos funcionales o del rendimiento. El mantenimiento del software se aplica a cada uno de los pasos precedentes del ciclo de vida a un programa existente en vez de a uno nuevo.

Como se puede apreciar, el paradigma clásico del ciclo de vida tiene un lugar definido e importante en el trabajo sobre ingeniería de software. Suministra una plantilla en la que puede colocarse los métodos para el análisis, diseño, codificación, prueba y **mantenimiento**.

1.5 Mantenimiento de Software

Los modelos de ciclo de vida tradicionales representan al mantenimiento como una fase que comienza una vez que se han finalizado las pruebas. Muchos estudios indican que es la fase más costosa del ciclo de vida del software. Algunas empresas han sobrepasado este porcentaje hasta llegar al límite de recursos, imposibilitando nuevos desarrollos, por ello se ha asegurado que el mantenimiento es la fase dominante del ciclo de vida⁵. Durante los últimos años se han propuesto diversas definiciones que identifican varias actuaciones para mantener la operatividad del software como:

- Corrección de defectos en el software
- Creación de nuevas funcionalidad en el software por nuevos requisitos de usuario
- Mejora de la funcionalidad y del rendimiento

⁵ Roger, S.Pressman "Ingeniería del software", 2d. Ed. McGraw-Hill, 1989

Algunas definiciones propuestas consideran sólo el primer aspecto como mantenimiento. A veces, los cambios en el software existente motivados por la ampliación de los requisitos originales se consideran proyectos de desarrollo continuado o de redesarrollo. Algunas organizaciones incluyen una actividad en el mantenimiento dependiendo de su esfuerzo o complejidad o según el personal que se va a encargar de realizarla. Estos aspectos subjetivos, son evitados en este trabajo, por lo que se considerará el mantenimiento en función de todas las actividades que se realizan una vez que se entrega el producto y es aceptado por el cliente.

Se define entonces, el mantenimiento como:

“el proceso de modificar un sistema o componente software después de su entrega, para corregir defectos, mejorar el rendimiento u otros atributos o adaptarlo a un entorno cambiante”.

1.5.1 Actividades y Tipos de Mantenimiento de Software.

¿Qué sucede respecto al mantenimiento o modificación de un sistema cuando el sistema no ha sido bien desarrollado? Necesitará cambios frecuentes por omisiones o nuevos requerimientos.

El mantenimiento de software es mucho más que una “corrección de errores”. Genera las siguientes actividades a través de las cuales puede tipificarse el mantenimiento.

La **primera actividad** de mantenimiento es cuando durante el uso de cualquier gran programa, se encuentran errores, siendo informado el equipo de desarrollo. El proceso que incluye el diagnóstico y la corrección de errores se denomina mantenimiento correctivo.

La **segunda actividad** se da cuando la vida útil del software se vuelve obsoleto para el entorno del sistema para el cual fue originalmente desarrollado, a esta actividad se le da el nombre de mantenimiento adaptativo.

La **tercera actividad** se da cuando a medida que se usa el software, se reciben recomendaciones de los usuarios sobre nuevas posibilidades de funciones ya existentes y sobre mejoras en general, con lo cual se le da el nombre de mantenimiento perfecto.

La **cuarta actividad** de mantenimiento se da cuando se cambia el software para mejorar una futura facilidad de mantenimiento, o fiabilidad. A menudo es denominado *mantenimiento perfecto*, esta actividad esta caracterizada por las técnicas de ingeniería inversa y de reingeniería.

En realidad, las tareas que se dan como parte del mantenimiento adaptativo o del perfecto son las mismas que se aplican durante la fase de desarrollo del proceso de ingeniería de software. Para adaptar o perfeccionar debemos determinar nuevos requerimientos, rediseñar, generar código y probar el software existente. Tradicionalmente, estas tareas han sido denominadas en conjunto como <<*mantenimiento*>>.

El mantenimiento de software ha sido hasta hace poco la caja negra del proceso de ingeniería de software. Los libros sobre mantenimiento contienen muy pocas referencias, en comparación con las fases de definición y de desarrollo del software. Se ha llevado a cabo poca investigación o producción de datos sobre el tema y se han propuesto pocos enfoques o “métodos” técnicos. De aquí que el objetivo de este trabajo sea crear una herramienta que a través de un protocolo guíe o auxilie a los responsables de dar un mantenimiento de software, a que el proceso sea un poco más fácil a través de un control estricto de documentos que favorezcan detectar desviaciones en sus procesos.

1.6 Problemas asociados al mantenimiento de software

La mayoría de los problemas asociados con el mantenimiento de software se deben a las deficiencias de la forma en que el software ha sido definido y desarrollado, aquí es donde aparece el síndrome del “pague ahora o pague después”. La falta de control y disciplina en el proceso de ingeniería del software casi siempre se traduce en problemas durante el mantenimiento, de aquí que la ingeniería sea un elemento clave para favorecer el mantenimiento.

Entre los muchos problemas clásicos asociados con el mantenimiento de software se encuentran los siguientes:

- A menudo es excepcionalmente difícil comprender un programa ajeno. A medida que existen menos elementos de configuración de software, mayor es la dificultad; si sólo existe el código indocumentado, es de esperar que aparezcan serios problemas.
- Ese personaje “ajeno” que realizó el programa a menudo no se encuentra alrededor para explicar el software.
- No existe una documentación apropiada o está mal preparada.

1.7 Facilidad De Mantenimiento

La facilidad de mantenimiento se puede definir cualitativamente como: facilidad para comprender, corregir, adaptar y/o mantener el software. La facilidad de mantenimiento es un fin clave que guía los pasos de cualquier metodología de ingeniería de software.

La facilidad de mantenimiento se ve afectada por muchos factores. Una falta de cuidado en el diseño, en la codificación o en la prueba tiene un impacto negativo sobre nuestra capacidad de mantener fácilmente el software, y probablemente el factor más importante que afecta, es la planificación condicionada al mantenimiento. Si se ve el software como un elemento del sistema que inevitablemente estará sujeto a cambios,

serán sustancialmente mayores las posibilidades de producir un software fácilmente mantenible.

1.8 Tareas de Mantenimiento

Las tareas asociadas con el mantenimiento del software comienzan con un plan, se prescriben procedimientos de evaluación y de información, se define una secuencia

estandarizada de sucesos para el mantenimiento y se establece un sistema de registro de información de las actividades de mantenimiento, además de definir criterios de revisión y de evaluación. Independientemente del tipo de mantenimiento, se siguen las mismas tareas técnicas. Estas tareas incluyen: modificación del diseño del software, revisión, modificaciones oportunas en el código, prueba de unidad y de integración, prueba de validación y revisión.

1.8.1 Registro de información

Históricamente, el registro de información de todas las fases del proceso de ingeniería del software ha sido inadecuado. El registro de información de mantenimiento ha sido prácticamente inexistente. Por esta razón frecuentemente no somos capaces de asegurar la efectividad de las técnicas de mantenimiento, incapaces de determinar la calidad de un programa y determinar lo que cuesta realmente el mantenimiento.

El primer problema que aparece con el registro de información es el de comprender qué datos merece la pena registrar. Swason⁶ da una extensa lista, la cual es utilizada para generar parte de la documentación necesaria para integrar en la carpeta de calidad:

1. Identificación del programa
2. Lenguaje de programación usado
3. Fecha de instalación del programa
4. Número de sentencias eliminadas en los cambios del programa

⁶ Swason, E.B. "The dimensions of Maintenance", Proc. 2nd Intl. Conf. Software Engineering, IEEE

5. Fecha de cambio del programa
6. Identificación del ingeniero de software
7. Tipo de mantenimiento
8. Fechas de comienzo y final del mantenimiento
9. Número de personas-hora acumulado del mantenimiento
10. Beneficios netos asociados con el mantenimiento realizado

La mayoría de los trabajos recientes sobre métricas del mantenimiento de software se centran en las características del software que más probablemente afectan a la frecuencia del mantenimiento y en modelos empíricos de predicción de la cantidad de trabajo de mantenimiento basándose en otras características del programa.

1.8.2 Documentación de los cambios a los programas

La modificación del software es peligrosa. En el área de sistemas es común oír la siguiente lamentación: “Pero si todo lo que hice fue cambiar esta sentencia...” Desgraciadamente, cada vez que se introduce un cambio en un complejo procedimiento lógico, la posibilidad de error aumenta. La documentación del cambio y una cuidadosa prueba de regresión ayudan a reducir los errores.

Cuando se usa en el contexto del mantenimiento de software, el término efectos secundarios implica un error u otro comportamiento indeseable aparecido como resultado de una modificación. Los efectos secundarios sobre la documentación se dan cuando no se reflejan los cambios del código fuente en la documentación del diseño y en los manuales orientados al usuario.

Siempre que se haga un cambio sobre el flujo de datos, sobre la arquitectura del diseño, sobre los procedimientos o sobre cualquier otra característica asociada, se debe actualizar la documentación técnica de soporte. La documentación del diseño que no refleja fielmente el estado actual del software es peor, incluso que la ausencia total de documentación. Cuando un examen inocente de la documentación técnica lleva a una determinación incorrecta de las características del software, se darán efectos

secundarios en los subsecuentes esfuerzos de mantenimiento. Para el usuario, el software sólo es tan bueno como lo sea la documentación que describe su uso.

Por ejemplo los mensajes de error no documentados pueden causar gran confusión; tablas, índices o texto no actualizados pueden llevar al usuario a la frustración y la insatisfacción.

1.9 Gestión de configuraciones de software para el mantenimiento

Los elementos que componen toda la información producida como parte del proceso de mantenimiento de los programas de software se denominan colectivamente configuración del software.

La gestión de configuraciones del software es un conjunto de actividades desarrolladas para gestionar los cambios a lo largo del ciclo de vida. EL GCS es una actividad de garantía de calidad de software que se aplica en todas las fases del proceso de ingeniería del software y aquí se ha utilizado para el caso del mantenimiento.

1.9.1 Líneas base

Los clientes desean modificar los requerimientos. El equipo de mantenimiento desea modificar el enfoque técnico. Los gestores desean modificar el enfoque del proyecto.

¿Por qué todas estas modificaciones? La respuesta es simple. A medida que pasa el tiempo, todo el mundo sabe más. Este conocimiento adicional es la fuerza motriz de la mayoría de los cambios y nos lleva a sentenciar algo que difícilmente aceptan los ingenieros de software: la mayor parte de los cambios están justificados.

Una línea base es un concepto de gestión de configuraciones del software que nos ayuda a controlar los cambios sin impedir seriamente los cambios justificados. Antes de que un elemento de configuración del software se convierta en una línea base, el cambio se puede llevar a cabo rápido e informalmente. Sin embargo una vez que se establece una línea base, se pasa figurativamente por una puerta de un solo sentido. Se pueden llevar a

cabo los cambios, pero se debe aplicar un procedimiento formal para evaluar y verificar cada cambio.

En el contexto del mantenimiento del software, se define una línea base como un punto de referencia en el cambio del software que queda marcado por el envío del elemento de la configuración del software y la aprobación. Por ejemplo, se documenta y se revisa un diseño preliminar de los cambios a los que serán sujetos los programas. Se encuentran errores y se corrigen. El diseño preliminar aprobado se convierte en una línea base. Sólo se pueden realizar cambios futuros en la arquitectura del software tras haber sido evaluados y aprobados.

1.9.2 Elementos de configuración de software

Un elemento de configuración del software (ECS) es un documento; un conjunto completo de casos de prueba o un componente de programa identificado (por ejemplo, un procedimiento en Pascal ó un paquete en ADA)

Los siguientes ECS son el objeto de las técnicas de gestión de configuraciones y forman un conjunto de líneas base:

1. Especificación del sistema
2. Plan del proyecto de cambio del software
3. a) Especificación de requerimientos de los cambios del software
b) Prototipo ejecutable o en papel
4. Manual de usuario preliminar
5. Especificación de diseño:
 - a) Diseño preliminar
 - b) Diseño detallado
6. Listados del código fuente

7. a) Planificación y procedimiento de prueba
b) Casos de prueba y resultados registrados
8. Manuales de operación y de instalación
9. Programas ejecutables
10. Manual de usuario
11. Documentos de mantenimiento
 - a) Informes de problemas del software
 - b) Peticiones de mantenimiento
 - c) Ordenes de cambios de ingeniería
12. Estándares y procedimientos de ingeniería de software

A continuación se examinan algunas de las tareas de GCS que se aplican a todos o algunos de los ECS anteriores.

- La principal responsabilidad de la gestión de configuraciones es el control de cambios. Sin embargo también es responsable de la identificación de los elementos individuales y de las distintas versiones de software, de las auditorías y de la generación de informes sobre los cambios realizados en la configuración.

Dentro del GCS se plantean cuatro tareas: identificación, control cambios, auditorías y generación de informes.

Identificación: Esta tarea de gestión de configuraciones del software asegura una disposición de nombres significativos y consistentes para todos los elementos de configuración de software. El término “disposición de nombres” se refiere a un esquema de identificación que proporciona la siguiente información:

1. Tipo de ECS (Documento, Programa, Caso de prueba,...)
2. Nombre del elemento de configuración (P. Ej., Especificación de diseño)
3. Identificación del proyecto o del producto
4. Número de versión

5. Fecha del último lanzamiento

Esta información puede ir codificada y localizada en un único código de identificación de ECS o puede ser referenciada como parte de información separadas. Estos datos de identificación para nuestro caso deben de ir en cada uno de los documentos generados que conformarán la carpeta de calidad.

Control de cambios.- Las modificaciones incontroladas llevan rápidamente al caos. El control de cambios, es la tarea de GCS más importante, ya que, proporciona un mecanismo para el control.

Primero se obtiene una solicitud de cambio y se evalúa para calcular el esfuerzo técnico, y los efectos secundarios, el impacto global sobre otras funciones del sistema y los costes estimados. Antes de que un ECS se convierta en una línea base, sólo es necesario aplicar un control de cambios formal y una revisión sobre las fases del mantenimiento que se llevará a cabo con el cliente. Una vez que el ECS ha pasado la revisión técnica formal y ha sido aprobado, se crea la línea base.

Una vez que se tiene la línea base, aparece el control de cambios a nivel proyecto. Ahora para hacer un cambio, el encargado del mantenimiento a ese programa debe recibir la aprobación del gestor del proyecto. En algunos casos se dispensa de generar formalmente las peticiones de cambio, los informes de cambio. Sin embargo, hay que evaluar la realización de cada cambio, seguir la pista y revisar todos los cambios.

RESUMEN

La ingeniería del software ofrece una clave para mejorar la productividad del mantenimiento. Con un cuidadoso diseño, una apropiada documentación y una serie completa de revisiones y de métodos de prueba, los errores serán fáciles de diagnosticar y de corregir cuando se den.

La gestión de configuraciones del software es una actividad de protección aplicada intensamente durante la fase de mantenimiento aunque también durante la definición y el desarrollo del software. La GCS identifica, controla, audita e informa de las modificaciones que invariablemente se dan al desarrollar el software una vez que ha sido distribuido a los clientes.

TÉRMINOS CLAVE

Mantenimiento de software

Ingeniería de software

Ciclo de Vida

CAPITULO 2

ADMINISTRACIÓN DE PROYECTOS DE MANTENIMIENTO DE SOFTWARE DE GESTION

Objetivos específicos: Mostrar las etapas propuestas del ciclo de vida para un proyecto de mantenimiento de programas de software de gestión y la documentación que se genera en cada fase. Definir tipos de proyectos de mantenimiento y establecer sus características.

Definición

Es Probable que la forma más antigua de trabajo organizado sean los proyectos. Las siete maravillas del mundo antiguo fueron productos elaborados individualmente, y aun hoy un gran porcentaje de actividades comerciales se realizan como proyectos individuales.

Los proyectos no se limitan a grandes obras de construcción, como represas, barcos o aviones. Escribir un ensayo, planear las vacaciones, preparar una campaña publicitaria o comprar una pieza de un equipo son también proyectos.

Pero cuál es el significado de la administración de proyectos de mantenimiento de software de gestión:

<< ...es la planeación y el control de varias actividades con el fin de corregir fallos en el código del software o adaptarlo por requerimiento de nuevas necesidades>>.

2.1 Tipos de proyecto de mantenimiento de software

Para su estudio y la construcción de la carpeta de calidad se han conformado dos tipos de proyectos de mantenimiento de software de gestión, de acuerdo al servicio que brinda cada uno.

Los tipos de proyecto que normalmente se manejan dentro del área de mantenimiento de software de gestión son:

- Proyectos de mantenimiento de software por función y tiempo
- Proyectos de mantenimiento de software por producto

Los proyectos de mantenimiento de software por <<**función y tiempo**>> son aquellos proyectos en los que se asignan recursos capacitados al cliente. El cliente asignará tareas de mantenimiento dentro de las horas establecidas en el contrato a cada uno de los recursos y él será el encargado de la supervisión, coordinación, prueba, e instalación del software modificado.

Los proyectos de mantenimiento de software por <<**producto**>>: son aquellos proyectos encargados de realizar un producto nuevo, debido a nuevo requerimiento de adaptación del software y los programas asociados a él. En este caso la administración de todo el proyecto esta a cargo del equipo de trabajo contratado y el cliente sólo se responsabiliza de validar los datos arrojados por dicho producto.

2.1.1 Características

Los proyectos de mantenimiento de software por función y tiempo tienen las siguientes características:

- Proporcionar recursos humanos especializados en mantenimiento de software de gestión a la organización del cliente
- Se asignan recursos individuales o grupos de trabajo con capacidades técnicas específicas para ser incorporados al cliente bajo su autoridad y dirección, es decir, la administración total del proyecto de mantenimiento esta a cargo del cliente.
- El personal es seleccionado con base a sus habilidades, capacitación y experiencia en el desempeño del trabajo requerido por el cliente

Los proyectos por producto tienen las siguientes características:

- Involucra el diseño de una solución de mantenimiento a la medida
- Propone, diseña, modifica e instala soluciones de negocio de acuerdo a las especificaciones de los clientes.

2.2 Ciclo de vida de los proyectos de mantenimiento de software.

Los proyectos son finitos: tienen un comienzo y un fin. En consecuencia, un proyecto tiene *ciclo de vida*. Las fases en las que se puede dividir el ciclo de vida son diferentes dependiendo del tamaño del proyecto o de la manera en que la empresa maneje la administración del proyecto, en este caso voy a utilizar un ciclo de vida de 4 fases. Estas son: Concepción, Formación, Operación y Terminación. Estas etapas se encuentran íntimamente involucradas con las fases de la administración; Planeación, Organización, Dirección y Control.

A continuación se describen cada una de las fases del ciclo de vida y la documentación que debe arrojar durante la administración del proyecto:

❖ Fase concepción

Durante la fase de concepción se estudia la idea de realizar un proyecto. Si es beneficioso y factible, la idea se transforma en una propuesta de proyecto, y luego se toma la decisión de “realizarlo” o no.

En esta fase se establecen los propósitos y se redacta *la propuesta*, en la que deben incluirse los beneficios esperados, los estimados de los recursos requeridos (personas, capital, equipo, etc.) y la duración estimada del proyecto (ver ejemplo de documento en capítulo 3, pág. 57)

En la propuesta se menciona como se abordará el problema y el tiempo que se requiere por cada fase del mantenimiento. La <<propuesta>> debe de contener

- Objetivos del proyecto
- Los acuerdos de confidencialidad
- Los recursos
- Las responsabilidades

Una vez aceptada la propuesta se pasa a la siguiente fase:

❖ **Fase formativa** (*Etapa de la Planificación y organización de proyectos de mantenimiento*)

Durante la fase formativa del proyecto se definen con claridad los objetivos, se selecciona el tipo de organización y se asigna el responsable del proyecto. La propuesta se transforma en un plan de proyecto maestro y se elaboran en detalle premisas, objetivos, estrategias, programas, requerimientos de recursos y presupuestos.

◆ **Premisas**

Las premisas son suposiciones que se deben de considerar ante aquellas circunstancias o condiciones futuras que afectarán el curso del proyecto.

Se establecen al inicio del proceso de planeación, aunque también pueden ser desarrolladas a lo largo del mismo.

◆ **Objetivos**

Una vez que se han establecido los propósitos e investigado las premisas que pueden afectar su realización, se determinan los objetivos, que indican los resultados o fines que

la empresa desea lograr en un tiempo determinado que proporcionan las pautas o directrices básicas hacia donde dirigir los esfuerzos y recursos.

Lineamientos para establecer objetivos

Aparte de que un objetivo debe siempre reunir las características de que se establezcan a un tiempo específico y se determinan cuantitativamente, al establecerlos es imprescindible observar ciertos lineamientos:

- Asentarlos por escrito.
- No confundirlos con los medios o estrategias para alcanzarlos.
- Al determinarlos, recordar las seis preguntas clave de la administración: ¿Qué, cómo, cuándo, dónde, quién y por qué?
- Los objetivos deben de ser perfectamente conocidos y entendidos por todo el equipo de trabajo.
- Deben de ser estables (es decir tener una *línea base*) los cambios continuos en los objetivos originan confusiones y conflictos.

◆ Estrategias

Las estrategias son curso de acción, que muestran la dirección y el empleo de los recursos y esfuerzos, para lograr los objetivos en las condiciones más ventajosas.

Lineamientos para establecer estrategias

Para que las estrategias sean operables y cumplan su función es necesario:

- Asegurarse de que sean consistentes y contribuyan al logro del objetivo.
- Determinarlas con claridad
- No confundir las estrategias con las tácticas, ya que estas últimas combinan la acción con los medios para alcanzar el objetivo.

- Considerar las experiencias pasadas para su establecimiento; Esto permitirá sugerir un mayor número de cursos de acción.
- Analizar las consecuencias que pudieran presentarse al momento de aplicarlas.
- Al establecerlas, auxiliarse e técnicas de investigación y de planeación.

◆ **Programas**

Una vez que se han determinado las anteriores etapas, es necesario elaborar un programa, el cual puede definirse como:

Un esquema donde se establecen: la secuencia de actividades específicas que habrán de realizarse para alcanzar los objetivos, y el tiempo requerido para efectuar cada una de sus partes y todos aquellos eventos involucrados en su consecución.

La elaboración técnica de un programa debe apegarse al siguiente procedimiento:

1. Identificar y determinar las actividades comprendidas.
2. Ordenar cronológicamente la realización de las actividades.
3. Interrelacionar las actividades. Determinar que actividad debe realizarse antes que otra
4. Asignar a cada actividad la unidad de tiempo de su duración, así como los recursos necesarios.

La elaboración de programas es fundamental, debido a que:

- Suministran información e indican el estado de avance de actividades
- Mantienen en orden las actividades, sirviendo como herramienta de control en los proyectos
- Identifican a las personas responsables de llevarlos a cabo, ya que se determina un programa para cada centro de responsabilidad.
- Determinan los recursos que se necesitan
- Determinan el tiempo de iniciación y terminación de las actividades

- Orientan a los programadores y analistas sobre las actividades que deben realizar específicamente.

Lineamientos para elaborar un programa

Para elaborar un programa hay que tener en cuenta las siguientes observaciones:

- Deben participar en su formulación: tanto el responsable del programa, como aquellos que intervendrán en la ejecución.
- La determinación de los recursos y el periodo para completarlo, deben estar relacionados con las posibilidades de la empresa.
- La aprobación del desarrollo y ejecución del programa, debe comunicarse a todos aquellos que estén involucrados en el mismo.
- Deben ser factibles
- Deben establecerse por escrito, graficarse y de fácil comprensión

◆ Presupuestos

El presupuesto es un plan de todas o algunas de las fases de actividad del equipo expresado en términos económicos, junto con la comprobación subsecuente de las realizaciones de dicho plan.

◆ Procedimientos

Los procedimientos establecen el orden cronológico y la secuencia de actividades que deben seguirse en la realización de un trabajo repetitivo.

Lineamientos para su implantación:

Para que los procedimientos cumplan su función, es necesario que:

- Sean accesibles y fáciles de interpretar
- Al implantarlos se adecuen a situaciones reales y factibles
- Deben adecuarse a las necesidades y características de cada proyecto
- No deben ser demasiado rígidos

El documento llamado plan de trabajo esta constituido por los siguientes elementos:

- Esfuerzos
- Recursos
- fecha inicio
- Fecha fin
- Costo

La elaboración y definición de todos los conceptos anteriores relacionado al proyecto de mantenimiento dan como resultado la generación de los siguientes documentos.

- **Acuerdo administrativo.** En este documento se establecen los términos y las condiciones base de referencia para una administración efectiva del proyecto.
- **Plan de actividades.** Se realiza un esquema mediante un diagrama de Gantt, de las actividades que se realizarán sin ser demasiado específicos y el tiempo aproximado para concluir las actividades.
- **Relación del equipo de trabajo.** Se indica el número de recursos que van a trabajar en el proyecto, su puesto, y sus obligaciones.
- **Matriz de habilidades.** Se establece si los recursos requieren algún tipo de capacitación y en que porcentaje cumplen con los conocimientos requeridos por el cliente.
- **Comunicaciones.** Se plantea la estructura de la organización, las comunicaciones y las jerarquías.

❖ **Fase Operativa** (Etapa de la organización)

En la fase operativa ya debe de estar conformado el equipo de trabajo. En este momento comienza las actividades del mantenimiento.

Realizar el proceso de seguimiento al progreso del proyecto, actualizar los planes del proyecto y vigilar de cerca el equipo son responsabilidades administrativas clave del líder del proyecto en esta etapa. Ocuparse de cambios propuestos en el trabajo por hacer o en los objetivos del proyecto es, quizá, la tarea más desafiante de todas.

Una vez que se ha establecido lo que se quiere hacer, será necesario determinar como hacerlo, qué medidas utilizar para lograr lo que se desea; esto sólo es posible a través de la organización.

En esta fase se generan los siguientes documentos:

- **Procedimiento de seguimiento al avance.** Este documento puede ser minuta ó enunciados resultado de juntas.
- **Reporte de actividades.** Este documento es elaborado por cada uno de los que participan en el proyecto, en el se establecen las actividades realizadas durante “x” periodo.
- **Plan de actividades del proyecto.** Se verifican actividades y tiempos del proyecto.
- **Configuración.** En este documento se relaciona todo lo que influya en tu producto que sea susceptible a los cambios necesarios para concluir con el

proyecto de manera satisfactoria. Los documentos que se incluyen en esta parte son: esquema de configuración y el control de versiones.

- **Control de asuntos.** En este documento se registran los problemas o incidentes ocurridos a lo largo del proyecto, los cuales se resuelven a nivel del responsable de proyectos. Por ejemplo: falta de energía, no hay sistema, etc.

- **Productos de trabajo.** Son todos los productos generados durante el proyecto y que son necesarios para integrar los entregables; ejemplo: manuales, código fuente, etc.
- **Mecanismo de almacenamiento y respaldo.** Aquí se menciona el procedimiento a llevarse a cabo para el respaldo y almacenamiento de los componentes modificados.

❖ **Fase de terminación** (Etapa de control administrativo)

En la fase de terminación ya se debe haber completado el trabajo en el proyecto (o suspendido) Durante esta fase se analizan los éxitos y fracasos del proyecto, se prepara un informe detallado para los equipos de proyectos futuros y se asignan nuevas tareas a los miembros del equipo.

Aunque los proyectos tienden a ser únicos en uno u otro aspecto, un minucioso examen posterior puede ayudar a los administradores. Un buen análisis y una buena comprensión crean conocimiento que es útil en este momento. Evitar los errores del pasado y aprovechar las formas organizativas mejoradas, las técnicas de planeación y control y los diversos estilos de administración ayudan a que las empresas diseñen y administren los equipos con mayor efectividad en el futuro.

En esta etapa se producen los siguientes informes:

- **Evaluación del proyecto por parte del cliente.** En este documento el cliente valora el resultado del proyecto a través de escalas establecidas por parte de la empresa que realiza el trabajo.
- **Evaluación de desempeño de los recursos asignados.** En este documento se juzga la participación de los recursos asignados al proyecto, mediante escalas de desempeño elaboradas por parte de la empresa donde colaboran los recursos.
- **Reporte de ejecución de desempeño.** Este documento incluye: metas logradas, problemas encontrados, Metas por cumplir.

- **Contrato.** El contrato deberá contener la firma del cliente, aceptando con esto la finalización del proyecto.

2.3 Diagramas de Gantt

Los cronogramas de barras o “gráficos de Gantt” fueron concebidos por el ingeniero norteamericano Henry L. Gantt, uno de los precursores de la ingeniería industrial contemporánea de Taylor, quien desarrolló el concepto en Estados Unidos a principios del siglo XX. Gantt procuro resolver el problema de la programación de actividades, es decir, su distribución conforme a un calendario, de manera tal que se pudiese visualizar el periodo de duración de cada actividad, sus fechas de iniciación y terminación e igualmente el tiempo total requerido para la ejecución de un trabajo. El instrumento que desarrolló permite también que se siga el curso de cada actividad, al proporcionar información del porcentaje ejecutado de cada una de ellas, así como el grado de adelanto o atraso con respecto al plazo previsto.

Estos diagramas son ayudas visuales y consisten en barras horizontales en las cuales la lista de actividades va debajo del eje vertical y las fechas se colocan a lo largo del eje horizontal.

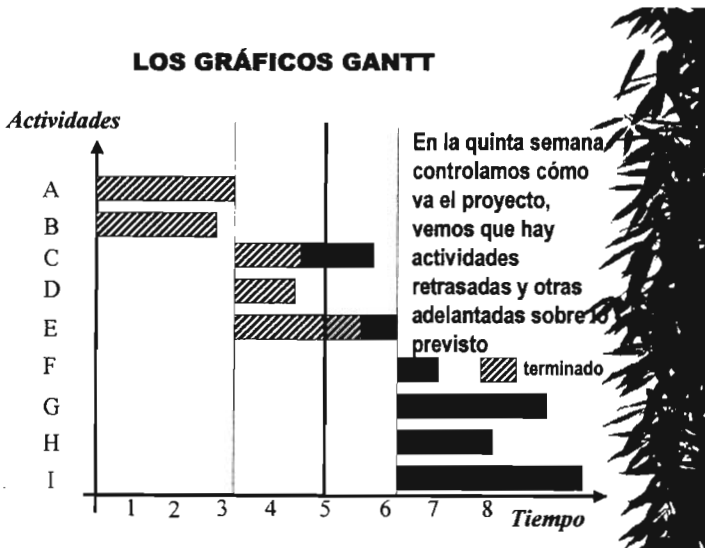
En un diagrama de Gantt, cada barra representa la duración de una actividad; el principio y final de la barra reproducen las fechas estimadas del comienzo y el final de la actividad.

Las actividades que comienzan más temprano se localizan en la parte superior del diagrama, y las que comienzan después se colocan de modo progresivo, la que empiece primero, en el eje vertical. De este modo, el diagrama parece la vista lateral de una corriente que fluye de una montaña, lo cual explica por qué los diagramas de Gantt también se conocen como diagramas en “cascada”. Además, el flujo desde la parte superior izquierda hacia la parte inferior derecha puede dar la idea de secuencia al colocar el número o la letra de la actividad precedente inmediata a la izquierda del extremo de la barra que representa la actividad.

El gráfico de Gantt se presta para la programación de actividades de la más grandes especie, desde la decoración de una casa hasta la construcción de una nave. Desde su creación ha sido un instrumento sumamente adaptable y de uso universal, dada su fácil construcción.

En el desarrollo de un proyecto es común que se disponga de recursos limitados para la ejecución de actividades. El gráfico de Gantt permite identificar la actividad en que se estará utilizando cada uno de los recursos y la duración de esa utilización, de tal modo que puedan evitarse periodos ociosos innecesarios y se dé también al administrador una visión completa de la utilización de los recursos que se encuentran bajo su supervisión.

A continuación se presenta un ejemplo de diagrama de Gantt, el cual fue elaborado con la finalidad de observar el avance de algún proyecto.



Nota del capítulo: Todos los documentos deben de contener: nombre del proyecto, Título del documento, página y fecha de elaboración.

RESUMEN

La administración es una herramienta que nos auxilia a llevar el control de nuestro proyecto, para detectar desviaciones y corregirlas a tiempo, con el fin de terminar dentro del tiempo y el costo estipulado con el cliente.

Los proyectos de mantenimiento tienen un ciclo de vida, que en este trabajo se manipulo a través de cuatro fases: concepción, formación, operación y finalización. Mediante estas fases se monitorea y se verifican los avances o desviaciones hacia los objetivos y se generan los documentos para la carpeta.

En la fase de concepción se parte de un requerimiento solicitado por el cliente, y se realiza la propuesta a la solución del problema. Una vez aceptada la propuesta se pasa a la siguiente fase: formación, aquí se lleva a cabo la definición del proyecto y se generan durante esta fase varios documentos entre ellos, el plan de actividades, el plan de calidad, el acuerdo administrativo y la matriz de habilidades entre otros.

Posteriormente se pasa a la fase operativa, que es donde se realizan las modificaciones a los programas, se generan los documentos de avances y se realizan las juntas de seguimiento. Finalmente en la fase de terminación se evalúan los recursos, se analizan los éxitos y los fracasos y se integran los últimos documentos en la carpeta de control de calidad.

Aunque los proyectos tienden a ser únicos en uno u otro aspecto el tener un componente estratégico, como lo es la carpeta de calidad puede servir de retroalimentación para evitar los errores del pasado y aprovechar los registros de proyectos exitosos anteriores. Santayana dijo una vez "... los que no pueden recordar el pasado están condenados a repetirlo".

TÉRMINOS CLAVE

Administración de proyectos de mantenimiento de software de gestión

Fase concepción

Fase formación

Fase de operación

Fase de terminación

Diagrama de Gantt

CAPITULO III

PROTOCOLO DOCUMENTAL PARA INTEGRAR LA CARPETA DE CALIDAD DE PROYECTO.DE MANTENIMIENTO DE SOFTWARE DE GESTION.

- Una estrategia para el seguimiento de los proyectos –

Objetivos específicos. Relacionar los conceptos de calidad y la norma ISO 9000 con el protocolo propuesto para la integración de la carpeta de calidad de proyectos de mantenimiento de software. Enunciar los requerimientos del protocolo y construir una carpeta para ejemplificar la documentación.

3.1 Definición de Administración de calidad

La administración de calidad es un conjunto de acciones que una organización realiza para mejorar y asegurar de manera consistente, la calidad que brinda a sus clientes. Un ejemplo de estas acciones son las que se llevan a cabo en los laboratorios químicos médicos donde existen una serie de pasos y procedimientos para evitar que el producto sea contaminado al momento de envasarlo en las líneas de producción.

Edward Deming un estadounidense, quién después de la segunda guerra mundial, enseñó a los líderes empresariales japoneses y ejecutivos de mas alto nivel, logro cambiar sus mentalidades y los convenció de que la calidad era una arma estratégica. Demostró los altos costos en que una empresa incurre cuando no tienen un proceso planeado para administrar su calidad, el costo de retrabajar dos o más veces los productos para eliminar los defectos. Deming hizo una comparación entre los costos de esfuerzos que representaban aplicar los principios del control de calidad, y demostró que las empresas podían obtener el doble de utilidades, si administraban correctamente.

Armand V. Feigenbaum, nacido en E.U.A, creador del concepto control de calidad, afirmó que decir calidad no significa “mejor”, sino, el mejor servicio y precio para el cliente.

3.1.1 Concepto general de la calidad

La calidad es el conjunto de cualidades, atributos y propiedades de una persona, objeto u organismo que satisfacen necesidades del cliente a través del cumplimiento de los requisitos establecidos. Actualmente la calidad se alcanza con base en la satisfacción de las necesidades de los clientes, así como de sus expectativas con productos y servicios competitivos.

La calidad consolida la confianza del cliente asegurando su fidelidad. En la búsqueda de la calidad surge la normalización a través de la cuál se establecen procesos para el mejoramiento de los productos y/o servicios.

Existen diferentes normas de calidad, pero para esta tesina se utiliza la norma ISO 9000, puesto que es una de las más reconocidas en el mundo.

3.2 ¿Qué es ISO?

ISO (International Organization for Standarization)⁷ es la entidad responsable para la normalización a escala mundial con una agrupación hasta la fecha de 91 países.

En nuestro país tenemos normas oficiales(NOM's), las cuales son regulaciones técnicas de observancia obligatoria expedidas por el Comité Técnico Nacional de Normalización de Sistemas de Calidad (COTENNSISCAL), el cual es responsable de la elaboración y revisión de las normas equivalentes a ISO 9000; su propósito es promover el desarrollo de la normalización para fomentar a nivel internacional el intercambio de bienes y servicios. En 1992 la serie (NOM's) cambia la nomenclatura a (NMX's) La Dirección

⁷ Organización fundada en 1946, con el fin de crear un conjunto común de normas para la manufactura, el comercio y las comunicaciones.

General de las Normas (DGN) de la Secretaría de Comercio y Fomento Industrial (SECOFI) es la representante en México.

3.2.1 Normas ISO 9000

La ISO 9000 es un conjunto de normas y directrices internacionales para la gestión de la calidad que desde su publicación inicial en 1987, han obtenido una reputación global como base para el establecimiento de sistemas de gestión de la calidad.

3.2.1.1 ¿Quién desarrolló la serie ISO 9000?

El comité técnico 176 (ISO/TC176) se formó en 1974 para armonizar la creciente actividad a nivel mundial en administración y aseguramiento de calidad. El subcomité 1 se estableció para la normalización de términos, lo cual dio como resultado la norma ISO 8402 en 1986 y el subcomité 2 emitió en 1987 las cinco normas que originalmente integraban la serie ISO 9000. En 1994 se emite la penúltima revisión a estas normas y hasta la fecha la serie 9000 se compone de casi 20 normas. La nueva familia se emite a partir de diciembre del 2000 y substituye a la versión 1994.

3.2.1.2 ¿En que consiste la serie ISO 9000?

La versión 1994 de ISO 9000 es una serie de normas que principalmente se dividen en normas guía y normas contractuales. Las normas guía sirven para aclarar algunos requisitos contenidos en las normas contractuales, que son aquellas que están sujetas a certificación y que se aplican especialmente cuando existe una relación entre 2 partes cliente y proveedor mediados por un contrato.

La nueva familia ISO versión 2000 consiste en las siguientes normas:

- ISO 9000:2000 Describe los principios y terminología de los sistemas de gestión de calidad.

- ISO 9001:2000 Especifica los requisitos para los sistemas de gestión aplicables a toda organización que necesite demostrar su capacidad para proporcionar productos o servicios que cumplan los requisitos de sus clientes. Su finalidad es la satisfacción del cliente. Esta es la única norma sujeta a certificación. Desaparece ISO 9002:1994 e ISO 9003:1994.
- ISO 9004:2000 Proporciona directrices que consideran tanto la eficacia como la efectividad del sistema de gestión de calidad. Su objetivo es la mejora en el desempeño de la organización.
- ISO 19011 Proporciona orientación relativa a las auditorías de gestión de la calidad y de gestión ambiental.

3.2.1.3 ¿Calidad en productos y servicios?

El origen del interés actual por la calidad se puede explicar recurriendo al estudio de la comercialización de los productos. En el mercado tan competitivo de hoy no basta con producir y distribuir masivamente productos o servicios, vender es lo importante y esto sólo se produce con la seguridad de la aceptación por parte del cliente. Así la calidad representa un concepto, apoyado en la satisfacción de los distintos tipos de compradores o, incluso, en los gustos de cada cliente individual⁸.

Pero ¿qué es la calidad? En la vida cotidiana, la calidad representa las propiedades inherentes a un objeto que permiten apreciarlo como mejor, igual o peor que otros objetos de su especie.

Los criterios en los cuales se basa el protocolo propuesto están basados en la norma de calidad ISO 9000, por esta razón se da a continuación la definición que proporciona la norma ISO 8402 sobre la calidad:

“Totalidad de características de un producto o servicio que le confieren su aptitud para satisfacer unas necesidades expresadas o implícitas”

⁸ Brian, Rothery. ISO 9000

En este caso se asume la definición de calidad del software como:

El grado con el que un sistema, componente o proceso cumple:

- Los requisitos especificados
- Las necesidades o expectativas del cliente o usuario.

3.2.1.4 Cláusulas de la norma ISO 9001

Existen varios modelos dentro de la norma ISO 9000 pero para el objeto de este trabajo se tomara el modelo de la norma ISO 9001 que se refiere al desarrollo, suministro y mantenimiento de 'software'.

En lo que a la norma se refiere el término "producto" incluye "servicios", "hardware", "software" y la combinación de los mismos.

La norma ISO 9001 tiene 20 cláusulas que establecen los documentos requisitos para certificar un sistema de calidad (aplicadas específicamente para el mantenimiento de programas de gestión). A continuación se mencionan los puntos de la norma y enseguida el documento obtenido durante el proyecto de mantenimiento de programas de software de gestión que se relaciona con la norma.

4.1 Responsabilidad de la dirección (política de calidad)

4.2 Sistema de calidad (Planeación de calidad)

4.3 Revisión del contrato (contrato)

4.4 Control de diseño/desarrollo (ciclo de vida del proyecto)

4.5 Control de documentación y datos (a nivel proyecto)

4.6 Compras (si se requiere algún tipo de software para dar el mantenimiento)

4.7 Control de productos proporcionados por el cliente (máquinas)

4.8 Identificación y rastreabilidad del producto (bibliotecas, índices, y ubicaciones de programas)

4.9 Control de proceso

4.10 Inspección y prueba (Definición de pruebas)

- 4.11 Control de los equipos de inspección, medición y prueba
- 4.12 Estado de inspección y prueba (registrar los resultados de las pruebas)
- 4.13 Control de productos no conformes
- 4.14 Acciones preventivas y correctivas (En el caso de ya se haya realizado alguna auditoria y existan acciones preventivas o correctivas)
- 4.15 Manejo, almacenamiento, empaque, conservación y entrega
- 4.16 Control de los registros de calidad
- 4.17 Auditorias internas de calidad
- 4.18 Capacitación y adiestramiento
- 4.19 Servicio post-venta
- 4.20 Técnicas estadísticas

Los documentos que se generan durante la administración de proyectos son registros de calidad los cuales son la evidencia objetiva del control de los procesos efectuados dentro del proyecto.

Si se necesita consultar alguna información de un proyecto concluido se puede buscar información en la carpeta de control de proyecto la cual deberá estar en poder del Gerente de negocio.

El alcance de aplicación de las cláusulas de la norma ISO 9001 a un proyecto varía de acuerdo a su línea de servicio. Las cláusulas que se aplican a cada línea son:

- *Proyectos por producto.* Para este tipo de productos aplican las veinte cláusulas de la norma y son las que se asumen para realizar el protocolo.
- *Proyectos por función y tiempo.* Para los proyectos por función y tiempo no aplican las siguientes cláusulas, ya que estos puntos son responsabilidad del cliente: 4.4 Control de diseño/desarrollo, 4.8 Identificación de productos y rastreabilidad, 4.9 Control de procesos, 4.11 Control de equipo de inspección y prueba, 4.12 Estado de inspección y prueba, 4.15 Manejo, almacenamiento, embalaje, preservación y entrega, 4.19 Servicio y la 4.20 Técnicas estadísticas.

3.3 Protocolo para la integración de la Carpeta de Calidad

El protocolo pretende ser un modelo o mejor dicho la pauta, para la construcción de una carpeta o portafolio que reunirá la documentación referente a la forma en la que se administra el proyecto de mantenimiento, el cual tiene como fundamento principal la retroalimentación y el control de desarrollo del proyecto.

El protocolo esta conformado por los puntos que se enumeran a continuación:

1. Designar un responsable para la integración de la carpeta (de preferencia el que funge como responsable del proyecto, ya que muchas veces se manejan datos confidenciales)
2. Establecer el ciclo de vida del proyecto a seguir. En este caso se tomo el modelo en cascada.
3. Realizar la documentación propuesta en el capítulo 2, a través de cada una de las fases durante la administración del proyecto.
4. Colocar la documentación elaborada en su sección y ubicación correspondiente (de preferencia puede colocar pestañas en cada sección para su localización rápidamente) de acuerdo a la guía que se localiza dentro del ejemplo en la página 47.

3.4 Beneficios y Limitaciones

Algunos de los beneficios del protocolo son los siguientes:

- Se establece una guía para la elaboración de una carpeta de calidad de proyectos de mantenimiento de software.
- Ahorra tiempo, al mostrar los documentos que se deben obtener en las fases del ciclo de vida del proyecto.
- Se ordena y sistematizada la documentación.
- Localización fácil y rápida de la documentación, gracias a su ubicación dentro de la carpeta de calidad de mantenimiento de proyectos de software
- El protocolo establece un mecanismo de retroalimentación para los siguientes proyectos

- Se puede validar a través de documentos el control que se tiene del proyecto.
- Se establece una memoria perene independiente del tiempo y de las personas

Limitaciones

- El protocolo esta basado en un ciclo de vida de 4 fases
- La norma ISO 9001-1998
- El protocolo establece ciertos pasos para la elaboración de la carpeta de calidad de proyectos de mantenimiento de software.

A continuación se muestra un ejemplo de la carpeta de calidad de proyectos de mantenimiento de programas de software de gestión.

**Carpeta de calidad de proyectos de mantenimiento de programas de
software de gestión**

INDÍCE

1. Inicio del Proyecto
 - 1.1 Plan de calidad
 - 1.2 Acuerdo administrativo
 - 1.3 Contrato

2. Administración del Alcance
 - 2.1 Propuesta
 - 2.2 Aceptación del proyecto
 - 2.3 Seguimiento al avance
 - 2.4 Evaluación de la satisfacción del cliente

3. Administración del trabajo
 - 3.1 Ciclo de vida y estándares
 - 3.2 Control de Asuntos
 - 3.3 Plan de trabajo
 - 3.4 Revisión al progreso

4. Administración del personal
 - 4.1 Equipo de trabajo
 - 4.2 Plan de capacitación
 - 4.3 Evaluación de desempeño

5. Administración de la organización y comunicación
 - 5.1 Comunicaciones
 - 5.2 Reportes de ejecución de desempeño

6. Administración de la configuración y productos de trabajo
 - 6.1 Configuración
 - 6.2 Mecanismo de respaldo
 - 6.3 Productos de trabajo
 - 6.3.1 Análisis
 - 6.3.2 Desarrollo del mantenimiento
 - 6.3.3 Pruebas
 - 6.3.4 Instalación

Sección 1	<i>Proyecto año 2000</i> PLAN DE CALIDAD	Ubicación CCP 1.1 Página: 1
-----------	--	--------------------------------

Administración:	Procedimiento o referencia	Ubicación del documento Dentro de la C.C.P	Responsable	Notas:
Inicio del Proyecto	Plan de calidad	1.1	Santiago Campos	
	Acuerdo administrativo	1.2		
	Contrato	1.3		
Del alcance	Propuesta	2.1	Santiago Campos	
	Acuerdo de Administración Secc. 5	1.2		
	Aceptación del producto de mantenimiento	2.2		
	Seguimiento al avance (Acuerdo de admon. Sec. 5)	1.2		
	Evaluación del cliente	2.4		
Del trabajo	Ciclo de vida (El ciclo de vida que se utiliza en esta empresa es el de cascada, Acuerdo admon.)	Estaciones de trabajo en las instalaciones del cliente.	Santiago Campos	
	Plan de trabajo	3.2		
	Revisión al progreso (Reportes de actividades)	3.3		
	Control de Asuntos (Acuerdo de administración Sección 6)	1.2 y 3.2		
Del personal	Equipo de trabajo	4.1	Santiago Campos	
	Plan de capacitación	4.2		
	Evaluación de desempeño	4.3		
De la organización y de la comunicación	Organigrama de comunicaciones (Acuerdo de administración secc.3)	1.2	Santiago Campos	

	Reportes de ejecución de desempeño	5.2		
Configuración y productos de trabajo	Esquema de configuración	6.1	Santiago Campos	
	Respaldos de programas (diskettes 3 ½)	6.2		
	Productos de trabajo (Puede colocar, los documento del lenguaje que haya utilizado para el proyecto)			
De riesgos	Lista de identificación y evaluación de riesgos (Cada responsable identificará los riesgos y su contingencia)		Santiago Campos	
	Plan de contingencia	7.2		
	Control de riesgos	7.3		

Sección 1	<i>Proyecto Año 2000</i> ACUERDO ADMINISTRATIVO	Ubicación 1.2 Página: 1
-----------	---	----------------------------

1. Introducción

El documento esta estructurado de la siguiente manera.

1. Introducción.- En esta parte se muestra como esta estructurado el acuerdo administrativo y presenta información con respecto al alcance y contenido.
2. Descripción del proyecto.- Presenta una manera generar el proyecto y sus objetivos.
3. Organización y comunicación.- En esta parte se establece el organigrama funcional de los grupos participantes en el mantenimiento, roles, responsabilidades y mecanismos de comunicación.
4. Modelo del ciclo de vida.- Especificar las fases y tareas del proceso de mantenimiento, así como los criterios de aceptación de los productos de trabajo.
5. Procedimientos de administración del proyecto.- Definir procedimientos y responsabilidades con respecto a la administración del proyecto, incluyendo:
 - ◆ Control de cambios al alcance
 - ◆ Procedimiento de aceptación de entregables;
 - ◆ Seguimiento al avance
 - ◆ Control de asuntos
 - ◆ Evaluación de la satisfacción del cliente
- 6.- Procedimientos de administración financiera.- Comunicar condiciones y procedimientos de aceptación de facturas.
- 7.- Comunicar acuerdos establecidos con respecto a recursos físicos y humanos.

2. Descripción del proyecto

Objetivo: El propósito de este proyecto es realizar el mantenimiento y preparar al sistema para el año 2000.

Exclusiones Esta empresa, no absorberá los impactos, provocados por la falta de energía, o en el retraso de otras partes que participen dentro del proyecto, y que sean ajenas a esta empresa.

Calidad Esta empresa, se compromete a llevar a cabo revisiones de los componentes modificados en su estructura y funcionalidad para asegurar la calidad de los productos entregados.

3. Organización y comunicación

El siguiente diagrama muestra la estructura de la organización y las líneas de autoridad de las diferentes funciones establecidas para los grupos de trabajo.



Comunicaciones. Se llevarán a cabo reuniones cada quince días o antes si fuese necesario para evaluar los avances con el Líder del cliente; esto con el fin de detectar desviaciones a los objetivos y planes del

mantenimiento; para establecer acciones correctivas. Otros medios de comunicación serán por medio de minutas o informes.

Equipo de trabajo

Funciones	Descripción
Líder de proyecto	Revisar, supervisar, implantar y evaluar todas las modificaciones realizadas, junto con el desempeño de los recursos.
Analista	Analizar y modificar los programas realizados en Cobol. Implantar las acciones correctivas identificadas. Realizar trámites para la instalación de los componentes modificados. Conocer y utilizar los estándares del cliente requeridas para la documentación del sistema.
Programador	Modificar los programas identificados según se le asigne. Apoyar en trámites e instalación de componentes.

Equipo de trabajo del cliente

Rol	Descripción de funciones
Líder de proyecto	Validar los productos modificados. Coordinar las sesiones de validación de los productos de trabajo. Convocar y dar seguimiento a las revisiones de avance del proyecto. Evaluar los cambios al contrato. Establecer los acuerdos y negociaciones con la diferentes áreas de negocio involucradas.

Plan de capacitación

Debido a que el equipo ya tienen conocimiento del sistema al cual fueron asignados no necesita ningún tipo de capacitación. (Ver matriz de habilidades, incluida carpeta control proyecto sección 3.2)

4. Modelo de ciclo de vida

Las fases del ciclo de vida se dividen de la siguiente manera:

Fase	Descripción
Fase 1. Diseño Funcional	Comprende las etapas de análisis preliminar y diseño conceptual de los componentes a modificar.
Fase 2. Diseño Técnico	Comprende etapas del diseño técnico de los componentes modificados.
Fase 3. Modificaciones	Comprende las modificaciones realizadas a los componentes
Fase 4. Pruebas Unitarias Prueba integral	Comprende las pruebas necesarias para validar el correcto funcionamiento de los componentes modificados.
Fase 5. Instalación	Comprende las actividades necesarias para la instalación de los componentes en el ambiente de producción, así como la observación de su comportamiento en dicho ambiente.

5. Procedimientos de administración del proyecto

Control de cambios

Si por algún motivo se presentan requerimientos cuyo impacto afecte al alcance y duración original del proyecto, estos serán analizados por los líderes del cliente y de esta empresa, para negociar los posibles cambios de duración e impacto financiero en el proyecto para verificar si son llevados a cabo.

Una vez que el usuario o líder por parte del cliente verificaron que los componentes modificados con los requisitos esperados, se procederá a firmar una carta de conformidad en la cual se especifique las condiciones de entrega, número, y custodia de los mismos, dicha carta puede ser la carta de aceptación de ese cliente.

Seguimiento al avance

Se efectuarán reuniones de seguimiento a las que asistirán el responsable del proyecto y el coordinador del área del banco. Las reuniones de seguimiento serán convocadas cada quince días o cada vez que así se requiera. Las minutas de esas juntas serán conservadas por el responsable del proyecto quien dará seguimiento a los compromisos que se establezcan.

Control de asuntos

El responsable del proyecto generará un reporte de la desviación, queja o cualquier otro incidente que cause impacto al proyecto y será entregado al Gerente de proyecto de nuestra empresa.

6. Procedimientos de administración financiera

Condiciones y procedimientos de aceptación de facturas.

Sección 1	Proyecto Año 2000 CONTRATO MAESTRO	Ubicación 1.3 30/10/1999 Página: 1
-----------	--	---

AI CONTRATO MAESTRO DE PRESTACION DE SERVICIOS DE
MANTENIMIENTO DE PROGRAMAS DE GESTION No. 589/Mant/ADDS

Nombre del proyecto:	AÑO 2000						
1.- Programa de cómputo mantenimientos	Modifica los programas del área de sistemas, reportes, mapas y tablas, para que acepten 4 dígitos. Sistemas secundarios de soporte en caso de que falle el acceso a DB2.						
2.- Especificaciones Técnicas	Soporte a programación, utilizando cobol, jcl, db2, cics						
3.- Tipo de Mantenimiento	A Sistema Central						
4.- Lugar de trabajo	Instalaciones del cliente						
5.-Objetivo	Dar soporte y mantenimiento al sistema por año 2000						
6.-Características y alcances	Se analizarán todos los programas que utilicen fechas en su operación y se modificarán. Se realizará un producto que interactúe con las entradas de datos de los programas en caso de existir contingencia.						
7.-Fases, etapas y actividades a que se sujeta el mantenimiento							
Fase	Producto Entregable	Fecha inicio	Fecha fin	Recursos	Tarifa por persona	Horas Por persona	Monto
01	Análisis	01/11/99	15/11/99	1 Analista 1Lider	\$ 215.00 \$ 280.00	88 88	\$18920.00 \$24640.00
02	Desarrollo (Cambios)	16/11/99	06/12/99	1 Analista 2 Programadores .50 Líder	\$ 215.00 \$ 189.00 \$ 140	120 120 60	\$20640.00 \$36288.00 \$ 6720.00
03	Pruebas	07/12/99	17/12/99	1 Analista 1 Programador	\$ 215.00 \$ 140.00	72 72	\$12040.00 \$ 7840.00
04	Instalación y soporte	20/12/99	11/01/2000	.50 Líder 1Programador	\$ 140 \$ 140	68 136	\$ 9520.00 \$ 19040.00
					Total:	716	\$151688.00
8.-Duración de los servicios				71 días			
9.- Total de horas del proyecto				716			
10.-monto de los honorarios				\$151688.00 (ciento cincuenta y un mil seiscientos ochenta y ocho pesos 00/100 M.N.)			
11.-Coordinadores y personal Clave				Cliente		Empresa	
				Lic. Alejandro Díaz Gómez		Ing. Santiago Campos	
12.-Criterios de aceptación				Toda actividad será revisada y aprobada por el supervisor a cargo del recurso.			
13.-Director de sistemas				Ing. Francisco Santoyo Mendez			

14.- Ejecutivo de cuentas	Lic. Alejandro Díaz Gómez
---------------------------	---------------------------

Este anexo se firma en la ciudad de México, Distrito Federal, a los días __ del mes de
de _____.

Empresa ADM.	Cliente
Análisis, Diseño, Desarrollo y Mantenimiento de sistemas SA de CV	XXXXXXXXXX
Firma	Firma

Sección 2	<i>Proyecto Año 2000</i> PROPUESTA	Ubicación 2.1 27/10/1999 Página: 1
-----------	--	---

PROPUESTA DE MANTENIMIENTO AL SISTEMA PARA EL AÑO 2000

Lic. Alejandro Díaz Gómez
 Ejecutivo de cuenta
 Empresa Bancaria
 PRESENTE.

Alejandro, como se acordó en la presentación de servicios del pasado día 25 de Octubre del presente año en tus oficinas, te envío nuestra propuesta económica para la prestación de Servicios de MANTENIMIENTO AL SISTEMA PARA EL AÑO 2000.

ANTECEDENTES.

Objetivo de la Propuesta Presentar los servicios que nuestra empresa ofrece para la conversión de los programas que contienen referencias de fechas.

Situación actual La empresa bancaria solicita la conversión de sus programas para que no se vean afectados con la llegada del año 2000.

Servicios que ofrecemos en general Dentro de los servicios que ofrece la empresa para esta propuesta son:

- Análisis de los programas que deben de ser modificados y de las relaciones que puedan resultar de dichas modificaciones.
- Modificaciones a los programas correspondientes y pruebas unitarias y modulares en las instalaciones del cliente.
- Trámites administrativos para la instalación en producción de los programas modificados .
- Soporte durante una semana a partir del 1º. De Enero del 2000.

SOLUCION PROPUESTA.

Solución
propuesta

Nuestra empresa plantea la siguiente propuesta de servicios para cubrir el requerimiento para el año 2000.

Para el análisis del sistema para ver que programas van a sufrir modificaciones se te proponen los siguientes Planes:

Etapas	Recursos	Tiempo	Costo
Análisis	1 Análista	2 semanas	\$17200.00
	1 Lider	2 semanas	\$22400.00
Mantenimient o Y pruebas	1 Analista	4 ½ semanas	\$32680.00
	2 Programadores	4 ½ semanas	\$44128.00
	.50 Lider	1 ½ semana	\$6720.00
Instalación y soporte	.50 lider	1 ½ semanas	\$9520.00
	1 Programador	3 ½ semanas	\$19040.00

En estos planes se tendrá la posibilidad de realizar otras modificaciones requeridas por parte del cliente, pero éstas sólo deberán ser en los programas que requieran la modificación del año 2000.

El pago del plan o los planes que te interesen se realizan al inicio del periodo.

Los costos de los planes se revisarán al inicio de cada periodo los cuales pudieran llegar a cambiar.

Todos los precios que se presentan en esta propuesta son en pesos más iva.

BENEFICIOS ESPERADOS

Se brindará soporte en sitio por parte de los analista durante la última semana de Diciembre y la primera semana de Enero del 2000 sin costo alguno, incluidos el sábado y el domingo.

Para el análisis del sistema para ver que programas van a sufrir modificaciones se te proponen los siguientes Planes:

Etapas	Recursos	Tiempo	Costo
Análisis	1 Analista	2 semanas	\$17200.00
	1 Lider	2 semanas	\$22400.00
Mantenimient o Y pruebas	1 Analista	4 ½ semanas	\$32680.00
	2 Programadores	4 ½ semanas	\$44128.00
	.50 Lider	1 ½ semana	\$6720.00
Instalación y soporte	.50 lider	1 ½ semanas	\$9520.00
	1 Programador	3 ½ semanas	\$19040.00

En estos planes se tendrá la posibilidad de realizar otras modificaciones requeridas por parte del cliente, pero éstas sólo deberán ser en los programas que requieran la modificación del año 2000.

El pago del plan o los planes que te interesen se realizan al inicio del periodo.

Los costos de los planes se revisarán al inicio de cada periodo los cuales pudieran llegar a cambiar.

Todos los precios que se presentan en esta propuesta son en pesos más iva.

TERMINOS Y CONDICIONES

Información Confidencial

Ambas partes se comprometen a proteger, manejar y utilizar toda la información, ya sea oral, escrita, visual o a través de un medio magnético o electrónico que reciban propiedad de la otra, única y exclusivamente para fines relacionados al proyecto, y a no divulgarla sin la autorización expresa del otro.

La información contenida en esta propuesta, no puede ser divulgada, duplicada y/o dada a conocer parcial o totalmente fuera de este centro bancario para propósitos diferentes a su evaluación por personal de dicha banca, sin la aprobación de esta empresa(con empresa me refiero a la empresa proveedora del servicio que realizo

esta propuesta).

Vigencia de la propuesta

La aceptación de esta propuesta anula la validez de cualquier otra con fecha anterior. Esta propuesta tiene una vigencia de 30 días naturales a partir de su fecha de recepción.

Sin más por el momento y esperando haberte mostrado que es de nuestro interés trabajar contigo, quedo en espera de cualquier comentario de tu parte y espero podamos ser favorecidos por tu decisión.

ATTE. Ing. Santiago Campos
Gerente de Área

c.c.p. Dir. Javier Barragán Almaraz.

Sección 2	<i>Proyecto Año 2000</i> CARTA DE ACEPTACION	Ubicación 2.2 11/01/2000 Página: 1
-----------	--	---

Nombre del proveedor:	Análisis, Diseño, Desarrollo y Mantenimiento de Sistemas S:A:
Contrato número:	589/Mant/ADDM
Proyecto nombre:	Proyecto Año 2000
Proyecto numero:	A2/25/2000

Descripción de los servicios brindados de conformidad con el cliente			
Descripción	Fecha inicio	Fecha término	Importe
Fase 01 Análisis	01-11-1999	15-11-1999	\$39600.00
Fase02 Mantenimiento	16-11-1999	15-12-1999	\$63648.00
Fase 03 Pruebas	13-12-1999	20-12-1999	\$19880.00
Fase 04 Instalación	22-12-1999	10-01-2000	\$28560.00
			SUBTOTAL
			\$151688.00
Comentarios:			IVA
			\$22753.2
			IMPORTE TOTAL
			\$174441.2
Importe total con Ciento setenta y cuatro mil cuatrocientos cuarenta y uno 25/100 letra: M.N.			
Prestación del Servicio:			
Servicio Nacional	<input checked="" type="checkbox"/>		
Servicio Internacional	<input type="checkbox"/>		

Recibimos los servicios descritos en esta carta y autorizamos el pago.

Ing. Verónica Nava
Subgerente del área de sistemas

Santiago Campos
Líder ADDM

Sección 2	Proyecto Año 2000 EVALUACIÓN DEL CLIENTE	Ubicación 2.4 09/01/2000 Página: 1
-----------	--	---

Compañía	Análisis, Desarrollo y Mantenimiento S.A.
Periodo:	01-11-1999/10-01-2000

	CUMPLIMIENTO			
	Deficiente	Inferior al adecuado	Bien	Excelente
	0-70	71-80	81-90	91-100
1.- Calidad del proyecto realizado				
2.- Conocimientos de los recursos asignados				
3.- Tiempo de entrega del proyecto				
4.- Generación de soluciones de valor agregado				
5.- Comunicación oportuna de riesgos				
6.- Actitud de servicio				
7.- Nivel de asesoría proporcionada				

Sección 3	<i>Proyecto Año 2000</i> CONTROL DE ASUNTOS	Ubicación 3.2 05/12/1999 Página: 1
-----------	---	---

		Solucionado(S) Pendiente (P)
Problema c/recurso		
Incidentes Medio Ambiente	Falta de energía ½ día, Sin sistema ½ día, Máquina con Virus 3 días	S
Otros:		

Sección 4	<p align="center"><i>Proyecto Año 2000</i></p> <p align="center">ORGANIZACIÓN DEL EQUIPO DE TRABAJO</p>	Ubicación 4.1 05/10/1999	Página: 1
-----------	--	-----------------------------	-----------

Nombre	Rol	Número de teléfono		
		Lugar del Cliente	Oficina	Casa
Berenice Montalvo R.	Programador	27143500	12458922	54785855
Alejandro Piña Sánchez	Programador	27143500	12458922	52235001
Carla Alvarez Santoyo	Analista	27143500	12458922	56784899
Santiago Campos Arce	Líder	27143500	12458922	56678945

Sección 4	<i>Proyecto Año 2000</i>	Ubicación 4.2
	MATRIZ DE HABILIDADES	05/10/1999 Página: 1

Requerimiento del cliente	COBOL	CICS	DB2
Capacidades del			
Berenice Programador	A	A	A
Alejandro Piña S. Programador	A	A	M
Carla Alvarez S. Analista	A	A	A
Santiago Campos A. Líder de proyecto	A	A	A

Parámetros:

A: ALTA
M: MEDIA
B: BAJA

Sección 4	<i>Proyecto Año 2000</i> EVALUACIÓN DE DESEMPEÑO	Ubicación 4.3 15/01/2000	Página: 1
-----------	--	-----------------------------	-----------

Nombre:	Alejandro Piña
Periodo:	01-11-1999/10-01-2000
Líder de proyecto	Santiago Campos

	CUMPLIMIENTO			
	Deficiente 0-70	Inferior al adecuado 71-80	Bien 81-90	Excelente 91-100
1.- Realización de lo asignado				
2.- Conocimientos Sobre los estándares del cliente				
3.- Generación de soluciones como valor agregado				
4.- Disponibilidad de tiempo				
5.- Sabe trabajar en equipo				
6.- Comunicación con su equipo de trabajo				
7.- Comunicación con el cliente				
8.- Actitud de servicio				

Sección 5	<i>Proyecto Año 2000</i> REPORTE DE EJECUCIONES	Ubicación 5.2 Página: 1
-----------	---	----------------------------

Gerencia de Negocio:	Sistemas
Responsable del proyecto:	Santiago Campos A.
Periodo del reporte:	10-nov-1999 al 10-ene-2000
Fecha de elaboración del reporte:	08-ene-2000

Requerimientos del Cliente:

Perfil Requerido	Recurso Asignado Nombre/Perfil	Fecha Requerida	Fecha de Asignación
Programador	Berenice	01-11-1999	01-11-1999
Programador	Alejandro Piña	01-11-1999	01-11-1999
Analista	Carla Alvarez	01-11-1999	01-11-1999
Líder	Santiago Campos	01-11-1999	01-11-1999

Desviaciones y Quejas

- Todos problemas, desviaciones, quejas que detectaron a lo largo del periodo que se está reportando.

Evaluación del Personal con Respecto al Proyecto.

Nombre Recurso	Desempeño (1 - 4)	Observaciones
Berenice	3	<i>Breve descripción de la percepción que el Responsable de Proyecto tiene con respecto a lo satisfecho que se encuentra el cliente con los recursos asignados y su desempeño.</i>
Alejandro Piña	3	
Carla Alvarez	4	

Desempeño : 1 = Deficiente 2 = Regular 3 = Bueno 4 = Muy Bueno

Revisión de Costos.

Planeado	Real
716 hrs.	\$151688.00

Revisiones de costos en el periodo.

Sección 6	<i>Proyecto Año 2000</i> CONTROL DE VERSIONES Y ESQUEMA DE CONFIGURACIÓN	Ubicación 6.1 10/12/1999	Página: 1
-----------	--	-----------------------------	-----------

Identificador del componente	Versión Actual	Ubicación Física	Fecha de Versión	Descripción del cambio
A2-MOD-01	3.0	A2_01/Modulos	Noviembre/99	Se cambiaron las opciones del menú
A2-MOD-02	4.1	A2_01/Modulos	Noviembre/99	Se modificaron parámetros de fechas
A2-MOD-03	2.5	A2_01/Modulos	Noviembre/99	Se modificaron mapas y rutinas
A2-MOD-04	5.8	A2_01/basededatos	Noviembre/99	Se modifíco campo de fecha
A2-MOD-05	1.6	A2_01/Manuales	Diciembre/99	Se documentaron las modificaciones
A2-MOD-06	4.3	X-Language	Diciembre/99	

Esquema de configuración

Nombre de componentes de configuración	Identificador único	Estado de liberación de primera versión	Número de versiones obsoletas retenidas
Modulo de pantallas del sistema	A2-MOD-01	Aprobado por el responsable del proyecto	1
Módulos de reportes del sistema	A2-MOD-02	Aprobado por el responsable del proyecto	1
Modulo de procesos del sistema	A2-MOD-03	Aprobado por el responsable del proyecto	1
Base de datos del sistema modificada para el proyecto	A2-MOD-04	Aprobado por el Cliente	0
Manual de desarrollo del sistema	A2-MOD-05	Modificación aprobada por el cliente	1

Lenguaje de desarrollo	A2-MOD-06	-	
------------------------	-----------	---	--

Conclusión

Por ser un componente guía para lograr un mejor control sobre la administración de los proyectos de mantenimiento de programas de software de gestión, el protocolo propuesto puede constituir un elemento táctico para ganar la batalla al tiempo, reducir costos y lograr un nivel de competitividad adecuada.

El protocolo establece a través de sus 4 pasos:

- ◆ Asignar un responsable del proyecto
- ◆ Definir etapas del ciclo de vida para el proyecto de mantenimiento
- ◆ Documentar durante las etapas del ciclo de vida
- ◆ Ordenar la documentación

una manera sencilla y organizada de archivar la documentación básica generada mediante la administración de proyectos de mantenimiento de programas de software de gestión, integrando con ellos un portafolio o carpeta a la que se ha llamado *carpeta de calidad*, mediante la cual se puede tener una visión general del proyecto de mantenimiento realizado.

Los documentos que conforman la carpeta contienen información y datos acerca de los problemas que se presentaron durante el desarrollo del mantenimiento, así como sus soluciones, lo que proporciona una retroalimentación para evitar cometer errores pasados en el futuro, además de preservar de manera perenne una memoria, independientemente del tiempo y de las personas que realizaron el proyecto. De tal manera que si se requiriera realizar un proyecto semejante o conocer de manera pormenorizada y fidedigna las actividades realizadas y sus consecuencias, siendo innecesario contar con el apoyo de las personas que participaron.

El índice incluido en la carpeta de calidad, representa una forma rápida para encontrar la documentación requerida en cualquier momento o tal vez durante una auditoría, por

ejemplo, si al abrir la pestaña correspondiente a *equipo de trabajo*; se encuentran los nombres de las personas que participaron, sus teléfonos y su perfil profesional y técnico, en fin toda la información correspondiente a esta sección, de manera semejante se puede encontrar el plan de trabajo establecido, los tiempos para cada actividad, etc..

Como en todo trabajo académico hubo obstáculos que se lograron superar, uno de ellos fue la obtención de datos reales, que se resolvió mediante el establecimiento de un proyecto ficticio y con base a este proyecto se generaron los documentos que sirvieron de ejemplo. Otro problema fueron las mínimas referencias teórico-conceptuales respecto al mantenimiento que se resolvió vía extrapolación de los conceptos manejados en la ingeniería de software sobre el ciclo de vida.

Con base a los conocimientos adquiridos en el transcurso de mi formación profesional en las materias de sistemas de información, confiabilidad, introducción a la administración y optimización, así como la experiencia obtenida durante mi participación en proyectos de mantenimiento trabajé para analizar la documentación que debía de integrarse en este tipo de proyectos, y a través de los conceptos de ingeniería de software y administración de proyectos establecer un ciclo de vida mediante la norma de calidad ISO 9000.

Después de resolverse los obstáculos y de realizar la investigación conceptual correspondiente se logró cumplir con el objetivo de este trabajo, al haber **establecido un protocolo para integrar una carpeta de calidad para la administración de proyectos de mantenimiento de programas de software de gestión**. Para terminar, debo reiterar que el protocolo propuesto no de manera alguna la panacea para el control de la documentación de los Proyectos de Mantenimiento de Programas de Software de Gestión mencionados, es exclusivamente un instrumento que puede auxiliar al líder del proyecto a coordinar y darle seguimiento el plan establecido con antelación, en el que estratégicamente, se aplica un principio de ordenamiento cronológico de documentación de cada una de las fases del ciclo de vida, al proceso de mantenimiento de programas de software de gestión.

ANEXO A

Aplicaciones del software:

Software de sistemas: es una colección de programas escritos para servir a otros programas. Algún software de sistemas (por ejemplo, compiladores, editores, utilidades y utilidades de gestión de archivos) procesan estructuras complejas de la información, aunque determinadas. Otras aplicaciones de sistemas (por ejemplo, componentes del sistema operativo, utilidades de manejo de periféricos) procesan datos en gran medida indeterminados. En cualquier caso, el área del software de sistemas se caracteriza por la fuerte interacción con el hardware de la computadora.

Software de tiempo real: es el software que mide/analiza/controla sucesos del mundo real conforme ocurren se llama de tiempo real. Los elementos del software incluyen una componente de acumulación de datos que recolecta y formatea la información de un entorno externo, una componente de análisis que transforma la información según requiera la aplicación, una componente de control de salida al entorno externo y una componente de monitorización que coordina a todas las demás componentes, de forma que pueda mantenerse la respuesta en tiempo real⁹.

Software de ingeniería y científico: se caracteriza por los algoritmos de “manejo de números”. Las aplicaciones van desde la astronomía a la vulcanología, desde el análisis de la presión de los automotores a la dinámica orbital de las lanzaderas espaciales y desde la biología molecular a la fabricación automática.

Software empotrado: Este software reside únicamente en la memoria de sólo lectura y se utiliza para controlar productos y sistemas de los mercados industriales y de consumidores. (por ejemplo: el control de las teclas de un horno de microondas)

⁹ El término tiempo real se diferencia del de “interactivo” o “tiempo compartido”. Un sistema en tiempo real debe responder dentro de unas ligaduras estrictas de tiempo. El tiempo de respuesta de un sistema interactivo (o de tiempo compartido) puede ser normalmente sobrepasado sin que se produzca ningún desastre.

Software de computadoras personales. El procesamiento de palabras, las hojas de cálculo, los gráficos por computadora, entretenimientos son sólo unos cuantos de los cientos de aplicaciones.

Software de inteligencia artificial: Este software hace uso de algoritmos no numéricos para resolver problemas complejos que no son adecuados para el cálculo o análisis directo.

ANEXO B

TIPOS DE MODELO DE CICLO DE VIDA

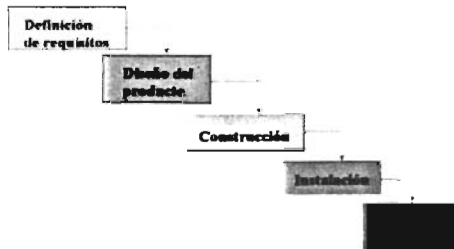
Las principales diferencias entre distintos modelos de ciclo de vida están en:

- El **alcance** del ciclo dependiendo de hasta dónde llegue el proyecto correspondiente. Un proyecto puede comprender un simple estudio de viabilidad del desarrollo de un producto, o su desarrollo completo o, llevando la cosa al extremo, toda la historia del producto con su desarrollo, fabricación, y modificaciones posteriores hasta su retirada del mercado.
- Las **características** (contenidos) de las fases en que dividen el ciclo. Esto puede depender del propio tema al que se refiere el proyecto (no son lo mismo las tareas que deben realizarse para proyectar un avión que un puente), o de la organización (interés de reflejar en la división en fases aspectos de la división interna o externa del trabajo).
- La **estructura** de la sucesión de las fases que puede ser lineal, con prototipado, o en espiral. Veámoslo con más detalle:

Ciclo de vida lineal

Es el más utilizado, siempre que es posible, precisamente por ser el más sencillo. Consiste en descomponer la actividad global del proyecto en fases que se suceden de manera lineal, es decir, cada una **se realiza una sola vez**, cada una se realiza **tras la anterior y antes que la siguiente**. Con un ciclo lineal es fácil dividir las tareas entre equipos sucesivos, y prever los tiempos (sumando los de cada fase).

Requiere que la actividad del proyecto pueda descomponerse de manera que una fase no necesite resultados de las siguientes (realimentación), aunque pueden admitirse ciertos supuestos de realimentación correctiva. Desde el punto de vista de la gestión (para decisiones de planificación), requiere también que se sepa bien de antemano lo que va a ocurrir en cada fase antes de empezarla.



Ejemplo de ciclo lineal para un proyecto de construcción

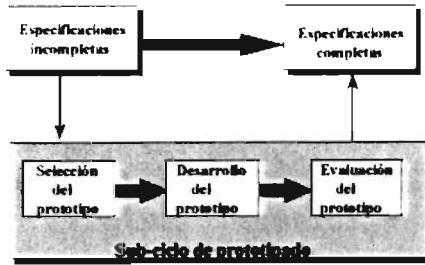
Ciclo de vida con prototipazo

A menudo ocurre en desarrollos de productos con innovaciones importantes, o cuando se prevé la utilización de tecnologías nuevas o poco probadas, que las incertidumbres sobre los resultados realmente alcanzables, o las ignorancias sobre el comportamiento de las tecnologías, impiden iniciar un proyecto lineal con especificaciones cerradas.

Si no se conoce exactamente cómo desarrollar un determinado producto o cuáles son las especificaciones de forma precisa, suele recurrirse a definir especificaciones iniciales para hacer un **prototipo**, o sea, un producto parcial (no hace falta que contenga funciones que se consideren triviales o suficientemente probadas) y provisional (no se va a fabricar realmente para clientes, por lo que tiene menos restricciones de coste y/o prestaciones). Este tipo de procedimiento es muy utilizado en desarrollo avanzado.

La experiencia del desarrollo del prototipo y su evaluación deben permitir la **definición de las especificaciones más completas y seguras** para el producto definitivo.

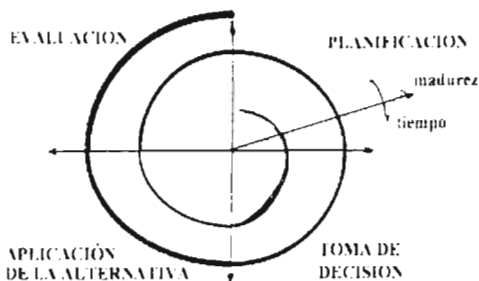
A diferencia del modelo lineal, puede decirse que el ciclo de vida con prototipado repite las fases de definición, diseño y construcción dos veces: para el prototipo y para el producto real.



Ciclo de vida en espiral

El ciclo de vida en espiral puede considerarse como una generalización del anterior para los casos en que **no basta con una sola evaluación de un prototipo** para asegurar la desaparición de incertidumbres y/o ignorancias. El propio producto a lo largo de su desarrollo puede así considerarse como una sucesión de prototipos que progresan hasta llegar a alcanzar el estado deseado. En cada ciclo (espirales) las especificaciones del producto se van resolviendo paulatinamente.

A menudo la **fuerza de incertidumbres es el propio cliente**, que aunque sepa en términos generales lo que quiere, no es capaz de definirlo en todos sus aspectos sin ver como unos influyen en otros. En estos casos la evaluación de los resultados por el cliente no puede esperar a la entrega final y puede ser necesaria repetidas veces.



El esquema del ciclo de vida para estos casos puede representarse por un bucle en espiral, donde los cuadrantes son, habitualmente, fases de **especificación, diseño, realización y evaluación** (o conceptos y términos análogos).

En cada vuelta el producto gana en "madurez" (aproximación al final deseado) hasta que en una vuelta la evaluación lo apruebe y el bucle pueda abandonarse.

OBJETIVOS DE CADA FASE

Dentro de cada fase general de un modelo de ciclo de vida, se pueden establecer una serie de objetivos y tareas que lo caracterizan.

Fase de definición (¿qué hacer?)

- Estudio de **viabilidad**.
- **Conocer los requisitos** que debe satisfacer el sistema (funciones y limitaciones de contexto).
- Asegurar que los **requisitos son alcanzables**.
- Formalizar el **acuerdo** con los usuarios.
- Realizar una **planificación** detallada.

Fase de diseño (¿cómo hacerlo? Soluciones en coste, tiempo y calidad)

- Identificar **soluciones tecnológicas** para cada una de las funciones del sistema.
- Asignar **recursos** materiales para cada una de las funciones.
- Proponer (identificar y seleccionar) **subcontratas**.
- Establecer métodos de **validación** del diseño.

- **Ajustar las especificaciones** del producto.

Fase de construcción

- Generar el producto o servicio pretendido con el proyecto.
- Integrar los elementos subcontratados o adquiridos externamente.
- Validar que el producto obtenido satisface los requisitos de diseño previamente definidos y realizar, si es necesario, los ajustes necesarios en dicho diseño para corregir posibles lagunas, errores o inconsistencias.

Fase de mantenimiento y operación

- **Operación:** asegurar que el uso del proyecto es el pretendido.
- **Mantenimiento** (nos referimos a un mantenimiento no habitual, es decir, aquel que no se limita a reparar averías o desgastes habituales –este es el caso del mantenimiento en productos software, ya que en un programa no cabe hablar de averías o de desgaste):

LOS PROYECTOS DE I+D

En el caso de la **investigación básica** el resultado esperado son conocimientos científicos. No existe ninguna fase de construcción y sí fases que recojan las tareas de experimentación.

En la **investigación aplicada** el resultado esperado suele ser alguna tecnología aplicable para procesos o para productos. Dependiendo del grado de cercanía a la aplicación que llegue a alcanzarse el modelo puede ser básicamente como el anterior o incluir una fase de aplicación piloto.

En el **desarrollo de productos o procesos** nuevos o significativamente modificados sí aparece ya una fase de construcción, aunque normalmente se tratará de la realización de un prototipo. Normalmente el cliente no será el usuario final, sino los departamentos de ingeniería de producción de la propia empresa o de otra que contrata el desarrollo.

La I+D es costosa por depender de personal muy cualificado, por realizarse de modo generalmente artesanal y por requerir bucles de realimentación que multiplican, para hacer frente a incidencias, la duración del proyecto.

REFERENCIAS BIBLIOGRAFICAS

- Rothery Brian (1993): ISO 9000. México: Panorama. Segunda Edición.
- Brookshear, J. Glenn (1993): TEORIA DE LA COMPUTACION. LENGUAJES FORMALES, AUTOMATAS Y COMPLEJIDAD. USA: Addison-Wesley Iberoamericana,S.A.
- Pressman, Roger S. (1993): INGENIERIA DE SOFTWARE: UN ENFOQUE PRACTICO, Madrid España:McGraw-Hill; Tercera edición.
- Cleland I. David; R.King William: MANUAL PARA LA ADMINISTRACION DE PROYECTOS (1999); Sexta Reimpresión; México; Compañía Editorial Continental, S.A. De C.V. Mexico
- Baca Urbina Gabriel (1995): EVALUACIÓN DE PROYECTOS McGraw-Hill, México. Tercera Edición
- J. Graham Robert; L. Englund Randall (1999) : ADMINISTRACION DE PROYECTOS EXITOSOS; Prentice Hall; México;
- Russel J.. P. (1998): EL PLAN MAESTRO DE CALIDAD;Panorama, México.
- Crosby, Philip B. (1979): QUALITY IS FREE; McGraw-Hill Book Company; USA
- Crosby, Philip B. (1991): LA CALIDAD NO CUESTA ; Continental, México.
- Guajardo Garza, Edmundo (1996) : ADMINISTRACION DE LA CALIDAD TOTAL; México:PAX.
- Fuente Ibarra, Ricardo de la (1985) : ADMINISTRACION E INFORMATICA: LA CIBERNETICA; México:ECASA.
- Johnson L. Perry (1993) : ISO 9000; McGraw-Hill México
- Martin Charles C. (1982) : ADMINISTRACION POR PROYECTOS; México:Diana.
- Martino R. L. (1965) : ADMINISTRACION Y CONTROL DE PROYECTOS; México: TECNICA.
- Drudis Antonio (1992) : PLANIFICACION, ORGANIZACIÓN Y GESTION DE PROYECTOS ; Barcelona: GESTION 2000.

REFERENCIAS DE DIRECCIONES ELECTRONICAS

<http://orbita.starmedia.unamosapuntes.tripod.com>

<http://www.adeit.uv.es/>

<http://www.labiblio.com/>

<http://aleph.biblioteca.uaem.mx>

**ESTA TESIS NO SALE
DE LA BIBLIOTECA**