

24021
2



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

ESCUELA NACIONAL DE ESTUDIOS
PROFESIONALES "ACATLÁN"

ANÁLISIS Y MEJORA DE LA PLANEACIÓN Y CONTROL DE
PROYECTOS DE UNA COMPAÑÍA DEDICADA AL
DESARROLLO DE SISTEMAS DE CÓMPUTO A LA MEDIDA

SEMINARIO-TALLER EXTRACURRICULAR
"EL ANÁLISIS DE LA PLANEACIÓN"

QUE PARA OBTENER EL TÍTULO DE

LICENCIADA EN MATEMÁTICAS APLICADAS Y
COMPUTACIÓN

PRESENTA

ROCÍO ALTAMIRANO HOLGUÍN

ASESOR: LIC. JUAN TORRES LOVERA



TESIS CON
FALLA DE ORIGEN

A

ABRIL, 2003



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.

PAGINACIÓN DISCONTINUA

Índice
NOMBRE: ROSA ALTAMIRANO
HAIGUN
2011
2011

Objetivo 4
Introducción 5

CAPÍTULO I

Objetivos de SERINF S.A de C.V y su planeación y control de proyectos 9
1.1 Breve historia de SERINF S.A de C.V 10
1.2 Visión y Misión 11
1.3 Servicios ofrecidos 12
1.4 El Sistema Local de Producción 17
 1.4.1 Infraestructura 19
 1.4.2 Catálogos de Trabajo 21
 1.4.3 Seguimiento de órdenes de trabajo 23
 1.4.4 Servicios 26
 1.4.5 Reportes 27
1.5 La Planeación y Control de Proyectos 29
 1.5.1 Estructuración del Proyecto 31
 1.5.2 Control de incidentes 32
 1.5.3 Mantenimiento del plan 36
Conclusiones 39

CAPÍTULO II

Análisis de las técnicas utilizadas para planear y controlar los proyectos 40
2.1 Análisis de la planeación en SERINF 41
 2.1.1 Tipo de Planeación 41
 2.1.2 Aplicación de planeación estratégica 58
 2.1.3 Proceso de estimación 54
2.2 Análisis del diseño y construcción de sistemas 60
 2.2.1 El diseño de sistemas 60
 2.2.2 Construcción de sistemas 62
Conclusiones 64

CAPÍTULO III

Propuestas de estrategias para reducir la incertidumbre 66
3.1 Servicio de diseño de sistemas 67
 3.1.1 El Lenguaje de Modelado Unificado (UML) 67
3.2 Servicio de construcción de sistemas 86
3.3 La planeación y control de proyectos 92
 3.3.1 Procesos maduros e inmaduros 93
 3.3.2 Aseguramiento de Calidad de Software 94
3.3.3 Monitoreo y control de proyectos de software 95
Conclusiones 97

TESIS CON
FALLA DE ORIGEN

Conclusiones generales.....	108
Glosario.....	111
Bibliografía	116

TESIS CON
FALLA DE ORIGEN

Dedico este trabajo a mis padres, quienes han sido un ejemplo de vida, han seguido de cerca, con confianza y entusiasmo todos mis pasos y me han dado su apoyo y cariño irremplazable e incondicional.

A mi hermana Maribel, porque la cercanía de un hermano es invaluable, porque juntas hemos pasado momentos realmente memorables y me enseñó a compartir y a escuchar a quien lo necesita.

A mis cuatro ángeles, Enrique A.F, Irene, Alfredo y Enrique A.H y a Sandy quienes de una u otra manera han influido en mi vida, esperan que siempre dé lo mejor... y que siempre estarán cerca de mí.

Al término de mis estudios profesionales, agradezco a Roberto, por apoyarme durante la final y a veces inalcanzable etapa, y demostrarme que todo es posible si se ama y se lucha con la fuerza necesaria.

A todos aquellos que me han ayudado en cualquier aspecto a lo largo de mi vida y durante el desarrollo de este trabajo: amigos, familiares, profesores, compañeros. Gracias por su preocupación y su tiempo.

Gracias, por último y no por menos importante, a Dios, por haberme dado la vida, enseñarme a valorarla, y sobre todo, haberme permitido compartirla al lado de las personas arriba mencionadas.

Objetivo General

Analizar la efectividad de la técnicas utilizadas para desarrollar y controlar los proyectos de una empresa proveedora de diseño y construcción de aplicaciones a la medida con el fin de actualizar las estrategias implementadas para reducir la incertidumbre en la confiabilidad, tiempos de entrega y en la diferencia de los costos reales contra las pronosticados inicialmente.

Introducción

Es un hecho que las empresas que quieren no solo ser competitivas, sino sobresalir de una manera notable, deben estar al día en cuanto a todo tipo de avances tecnológicos, renovación de modelos de administración y a lo que normas de calidad se refiere, así como estar informados sobre la situación económica, política y social mundial sea cual sea el área o sector de la industria en la que se desarrollen. Para tomar en cuenta todos estos aspectos y poder manejar de acuerdo a ellos una organización, es esencial la *planeación*, y es algo de lo que México adolece, no solamente hablando de organizaciones públicas, sino también en empresas privadas y en el estilo de vida en general.

Servicios en Informática y Desarrollo de México, S.A de C.V (SERINF, S.A de C.V), la fábrica de software sobre la que se basa este trabajo, y conocida comercialmente como *certum*, ha reconocido desde sus inicios tomar en cuenta esta área del conocimiento y ha sido una de las razones por la que no solo se ha mantenido en el mercado por 16 años sino que ha sobresalido como uno de los más destacados proveedores mexicanos de diseño y construcción de aplicaciones a la medida.

El factor que la dio a conocer internacionalmente fue su nivel de productividad, determinado en 1996 por el estándar internacional para evaluación de procesos de software "Software Process Improvement and Capability dEtermination" (SPICE). SPICE califica a las empresas del ramo en un rango de 0 a 5 y *certum* fue la primer organización mexicana en obtener un nivel de 3.76. Otro aspecto determinante que ha impulsado el avance de la compañía, es la explotación de técnicas de ingeniería de software avanzadas como el conteo de puntos de función (FP por sus siglas en inglés), métrica de software por medio de la cual se puede llevar a cabo la medición del avance real de construcción y del tamaño de los productos terminados.

Desgraciadamente los resultados de la empresa en los últimos dos años no han sido tan buenos como pudieran serlo, y la más reciente evaluación interna SPICE, realizada por el director general, arrojó un resultado de nivel 2 de productividad, con características de niveles 3 y 4. Este decremento ha provocado una preocupación general dado que era el factor que diferenciaba a la empresa del resto, y que podía mantenerla compitiendo con proveedores extranjeros. Pero gracias a que esta evaluación interna se realizó oportunamente, *certum* está muy a tiempo para poder ubicar los factores que han provocado esta situación, y así proponer soluciones.

De esta manera surge el motivo para la realización de este trabajo, que partiendo de la política interna de calidad que dicta "reducir la incertidumbre en la confiabilidad, en los tiempos de entrega y en la diferencia de los costos reales contra los pronosticados inicialmente", tiene como objetivo analizar y mejorar la planeación y control aplicados a los proyectos, así como las técnicas implementadas para llevar a cabo el diseño y la construcción de los sistemas.

Dicho de otra forma, el presente es un trabajo de análisis de la situación por la que está atravesando una empresa dedicada al diseño y construcción de aplicaciones de software a la medida. La metodología que se sigue consiste en:

1. Observación. Se requiere de un contexto que permita apreciar el giro de la empresa, sus inicios, cómo ha alcanzado sus logros, el tipo de clientes con los que cuenta, su misión y visión como organización.
2. Análisis. De la observación realizada destacan los procesos que han sido definidos para brindar cada uno de los servicios ofrecidos por certum, pues constituyen la base y el camino a seguir para contar con el producto final. Dichos procesos han sido analizados con el fin de identificar los puntos débiles sobre los que hay que trabajar.
3. Detección de problemas. Con base en los resultados obtenidos en el análisis, se enumeran los principales puntos causantes de incertidumbre en la calidad, fecha de entrega y costo de los sistemas.
4. Propuestas de mejoras. Una vez definidas las causas, se proponen soluciones basadas en las técnicas de software que mejor se adapten a las necesidades de la empresa.
5. Aseguramiento de la calidad. De nada sirve proponer soluciones si éstas no pasan de ser buenas intenciones a hechos palpables. Dado que la naturaleza del problema requiere de un largo tiempo para llegar a implementar las mejoras, y aún más para obtener resultados, esta investigación no abarca la experimentación de las propuestas; en su lugar se incluyen técnicas de auditorías que garanticen el seguimiento de procesos y así, contar con un factor que asegure el cumplimiento del objetivo principal: reducir la incertidumbre.

Los pasos anteriores son cubiertos a la largo de los tres capítulos de los que consta la investigación, de la siguiente manera:

En el capítulo 1, se da comienzo presentando un poco de la historia de certum, su misión, visión y la estructura del Plan de Proyectos implementado hasta el momento, destacando los puntos más importantes de la organización y que son llevados a cabo en todos o la mayoría de los proyectos según como sean aplicables, pues es del conocimiento de los encargados de producto, que los

sistemas no pueden ser tratados todos de la misma manera sino que dependerá de su giro y del contrato que se establezca con el cliente.

En la parte media del trabajo (capítulo 2), se realiza el análisis no solo de la planeación y control de proyectos, sino también de los procesos de diseño y construcción de sistemas; dicho análisis está basado en la forma como se han aplicado las distintas teorías y técnicas de planeación en el proceso de producción. Esta información se obtuvo de una base de datos interna que lleva el nombre de Diseño de Servicios®, en la cual se puede encontrar toda la documentación sobre políticas, estándares, servicios y procesos administrativos implementados.

Como se podrá apreciar después de la lectura del capítulo 2, la planeación empresarial ha sido un aspecto que se ha estudiado, implementado y actualizado cuidadosamente, pero convendría aplicar más rigor en el control del seguimiento del mismo y aplicar técnicas de planeación específicas para el desarrollo de software.

En la última parte entonces, se encontrará un apartado (3.3) que habla sobre la importancia de las auditorías internas de calidad que se encarguen de asegurar que los procesos asentados en los distintos documentos que conforman el Diseño de Servicios®, se cumplan en todas sus etapas. Este es un aspecto de suma importancia, pues una de las muchas bondades del cuidado de la calidad, es la garantía del máximo aprovechamiento del tiempo localizando problemas y riesgos oportunamente. Se encontrarán también propuestas de actualización de las técnicas y herramientas utilizadas para diseñar y construir software.

Conviene puntualizar, que para la perfecta comprensión del apartado 3.1, se recomienda que el lector cuente con un contexto sobre diseño de sistemas, programación estructurada y programación orientada a objetos, pues se hace referencia a conceptos sobre estas áreas, sin detenerse en definiciones, pues la literatura acerca de ellas es vasta y la intención es no desviar la atención del objetivo de la investigación, con marcos teóricos demasiado amplios (para referencia rápida, utilizar el glosario anexo). Las propuestas de construcción del apartado 3.2 otorgan especial atención a los aspectos que brinden un sencillo mantenimiento, pues éste se ha vuelto constante en los sistemas informáticos y es provocado por diversos factores como: el aumento del volumen de operación de las empresas, la necesidad de automatización de nuevos tipos de procesos, deseos de incursionar en el comercio vía Internet, entre otros.

El factor visionario toma gran importancia, pues para poder diseñar y construir productos robustos y escalables, es necesario tener lo más claro posible las necesidades presentes y futuras del cliente y poder ofrecerle lo que espera: la presentación transparente de la planeación y del avance de la construcción de los productos, con la certeza de que al terminar, habrá pagado un precio justo por

una aplicación que pueda ser puesta en marcha en el momento esperado y que permita actualizaciones sin demasiadas complicaciones, minimizando tiempos y costos.

Se espera que con la aplicación de las mejoras incluidas en esta investigación, se logre devolver a la empresa el nivel de productividad reconocido hace ya 6 años, e incluso rebasarlo en un rango de 3 a 5 años.

CAPÍTULO I

Objetivos de SERINF S.A de C.V y su planeación y control de proyectos

Objetivo: *En esta primera parte se desarrollará, de manera breve, la historia de la compañía, su misión, visión y su esquema de trabajo actualmente utilizado para llevar a cabo la planeación y control de proyectos de diseño y construcción de sistemas de cómputo.*

1.1 Breve historia de SERINF S.A de C.V

La compañía nació en Puebla en 1986, bajo el nombre de Servicios en Informática y Desarrollo de México. En esas fechas se introdujo al mercado el concepto de *Sistemas locales* (SISLOCS), los cuales eran sistemas que atendían un servicio a la vez, que podían integrarse en un conjunto de SISLOCS soportados por una red de servicios estratégicos. "Era la época en la que las redes locales nacían y Servicios en Informática desarrollaba sistemas grandes, para empresas grandes, en máquinas pequeñas, ya que los sistemas se desarrollaban en computadoras personales (PC's) y redes locales (LAN's - Local Area Network)"¹.

En 1988 la empresa cambia su domicilio a la ciudad de México ya que sus clientes (Primex, Conдумex, Salinas y Rocha) se encontraban en esta ciudad. Durante los siguientes años la empresa continúa desarrollando sistemas locales haciendo uso de la tecnología abierta, y es en 1991 cuando empieza a desarrollar sistemas en arquitectura cliente/servidor, en proyectos financiados por Publicaciones CITEM y Grupo casa Autrey. Paralelamente se diseñó y construyó un sistema para una orden religiosa internacional, modelando los procesos y haciendo uso de correo electrónico, administrando conversaciones y comunicando información entre varias ciudades del mundo; hasta la fecha es uno de los clientes más importantes de certum.

En 1994 SERINF comienza a participar en un proyecto de la ISO llamado SPICE (Software Process Improvement and Capability dEtermination). Este es el nombre de un estándar internacional para evaluación de procesos de software. El director general, Arnoldo Díaz Olavarrieta, fue nombrado ISO Project Editor y miembro del consejo de administración del proyecto SPICE; la herramienta de evaluación de procesos de software desarrollada por esta empresa fue denominada Official SPICE Assessment Instrument. Este proyecto ha dado como resultado el estándar ISO 15504 Software Process Assessment.

certum, el nombre de la marca, nace en 1997. El nombre obedece a una resolución que tomó la empresa de atacar un problema fundamental en la industria del desarrollo de software: la incertidumbre. Incertidumbre en la confiabilidad, en los tiempos de entrega y en la diferencia de los costos reales contra los pronosticados inicialmente.

También en 1997 certum obtuvo dos certificaciones ISO-9001 y una evaluación Bootstrap gracias al estándar ISO-15504, siendo la primera empresa en Latinoamérica en obtenerlo y con un resultado de 3.75. Hasta ese momento, en Estados Unidos solo el 9% de las empresas del ramo habían conseguido ese resultado, y en Europa, el 6%.

La compañía se ha mantenido hasta la actualidad gracias al nivel competitivo que ha logrado mediante el desarrollo de diseños de planeación y control de sus proyectos, tomando en cuenta siempre mantener informado al cliente del avance de los mismos. certum es reconocida como una de las empresas precursoras en el registro del reporte diario de actividades de sus ingenieros, mismo que se explicará con detalle más adelante, así como de la aplicación de técnicas de medición de software, como lo es el conteo de puntos de función.

1.2 Visión y Misión

Tanto la misión como la visión de la empresa están publicados en el sitio web www.certum.com, así como en una base de datos cuya consulta está disponible para todo el personal y que lleva como nombre "Servicios certum".

VISIÓN: Ser la mejor empresa de proyectos de ingeniería de software en México para los mejores clientes y los mejores ingenieros.

- *de ingeniería de software*, porque no solo está presente el compromiso de la entrega de los proyectos, sino que el proceso bajo el cual se construyan debe estar actualizado haciendo uso de la tecnología más avanzada en el ramo de la ingeniería de software.
- *en México*, porque la compañía se encuentra ubicada en la ciudad de México. Se cuidó de no especificar "de México", pues esto significaría que la mejor empresa estuviera fuera de México.
- *para los mejores clientes*, porque certum es selectivo en el aspecto de aceptar la realización de proyectos. Se cuida la imagen de la empresa trabajando para clientes serios, confiables y concientes de que el tipo de servicios ofrecidos por certum implica un costo cuyos beneficios pueden ser visualizados en el largo plazo.
- *y los mejores ingenieros*, pues el personal contratado por la empresa ha cumplido con exámenes que comprueban su capacidad para el análisis de algoritmos así como de su conocimiento de herramientas para el diseño y construcción de software.

MISIÓN:

"Garantizar la implementación exitosa de servicios estratégicos que hacen uso creativo de tecnología de información, con la menor inversión".

¹ www.certum.com

Así como en la visión se indica que se espera trabajar para los *mejores clientes*, es compromiso entonces de certum, cumplir con lo prometido minimizando la incertidumbre y maximizando la confianza que el cliente deposita en la empresa; esto se logra haciendo visible la utilidad que le proporcionará su inversión en software de calidad.

La visión y misión dependen del uso de tecnología avanzada, pero se recalca no solo su "uso", sino su "uso creativo", pues para estar a la vanguardia se deben conocer a fondo los recursos y explotarlos de manera óptima.

1.3 Servicios ofrecidos

certum ofrece los servicios de:

- Diseño de sistemas
- Construcción de software
- Pruebas formales

Dichos servicios son ejecutados a través de la red de servicios certum (diagrama 1.3.1).

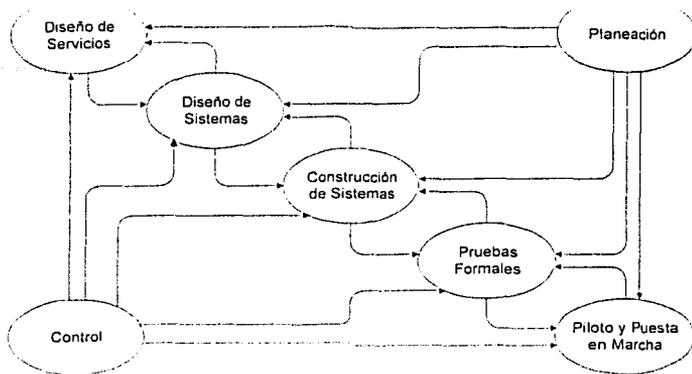


Diagrama 1.3.1 Red de servicios

A partir del desarrollo de estos servicios se desprenden los siguientes conceptos, los cuales son llevados a cabo para controlar los proyectos.

Análisis de Operaciones

El objetivo del Análisis de Operaciones es identificar los principales procesos a ejecutar por parte de la aplicación que se genere. Junto con los procesos, se detallan los procedimientos, que son la secuencia de pasos para ejecutar los procesos. Se identifican los volúmenes de datos y de transacciones, así como los estándares de operación del cliente. Si se requiere interactuar con alguna herramienta, se prevén las interfaces necesarias.

Diseño de Aplicaciones

El objetivo es diseñar la mejor aplicación, siendo ésta la que mejor apoye el servicio que se planea dar. Se diseña el producto de software, es decir, las interfaces de entrada y salida, la secuencia de operaciones a seguir dentro de la aplicación, el esquema de seguridad, considerando la plataforma tecnológica de hardware, software y comunicaciones. De esta forma se diseñan aplicaciones que "operan el negocio".

Pruebas Formales

Todos los códigos son probados en sus diferentes fases de construcción e Integración. Se verifica que los productos funcionen correctamente.

Piloto y Puesta en Marcha

Se busca la asimilación completa del software por el usuario, no solamente de las aplicaciones sino también de las técnicas y procesos. Para lograr esto se llevan a cabo actividades de capacitación para los usuarios finales, y de seguimiento del uso del proyecto, de manera que los usuarios detecten posibles errores de operación o que sugieran nuevos requerimientos para mejorar algunas operaciones.

Aseguramiento de Calidad

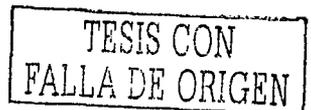
Hay una actitud general, expresada en políticas, para asegurar la calidad de todos los productos. Estas políticas están asentadas en procesos y procedimientos y el área de Aseguramiento de Calidad verifica con auditorías que siempre se ejecuten ambos. El enfoque que hay en la empresa de mejora y aseguramiento de calidad es a través del proceso, por lo que constantemente se analizan los resultados de cada fase del proceso y se verifica que se promuevan mejoras con base en esos análisis.

El objetivo de Aseguramiento de Calidad es revisar todos los trabajos antes de que se den por terminados. No se puede empezar ningún trabajo antes de que el anterior esté aprobado. Se hacen inspecciones formales a los diseños y se realizan pruebas formales a los productos. En ambas actividades se verifica tanto que el producto funcione correctamente como que satisfaga las necesidades del cliente.

Administración de Proyectos

certum tiene una aplicación propietaria de control y costeo denominada SILPRO (Sistema Local de PROducción). Basándose en el reporte diario de cada Ingeniero se puede evaluar el estado del proceso así como el costo del proyecto. Este análisis genera estados financieros por proyecto.

A continuación se presentan los diagramas de proceso que actualmente se siguen al llevar a cabo los servicios de diseño y construcción de sistemas, mismos que serán analizados en el capítulo siguiente.



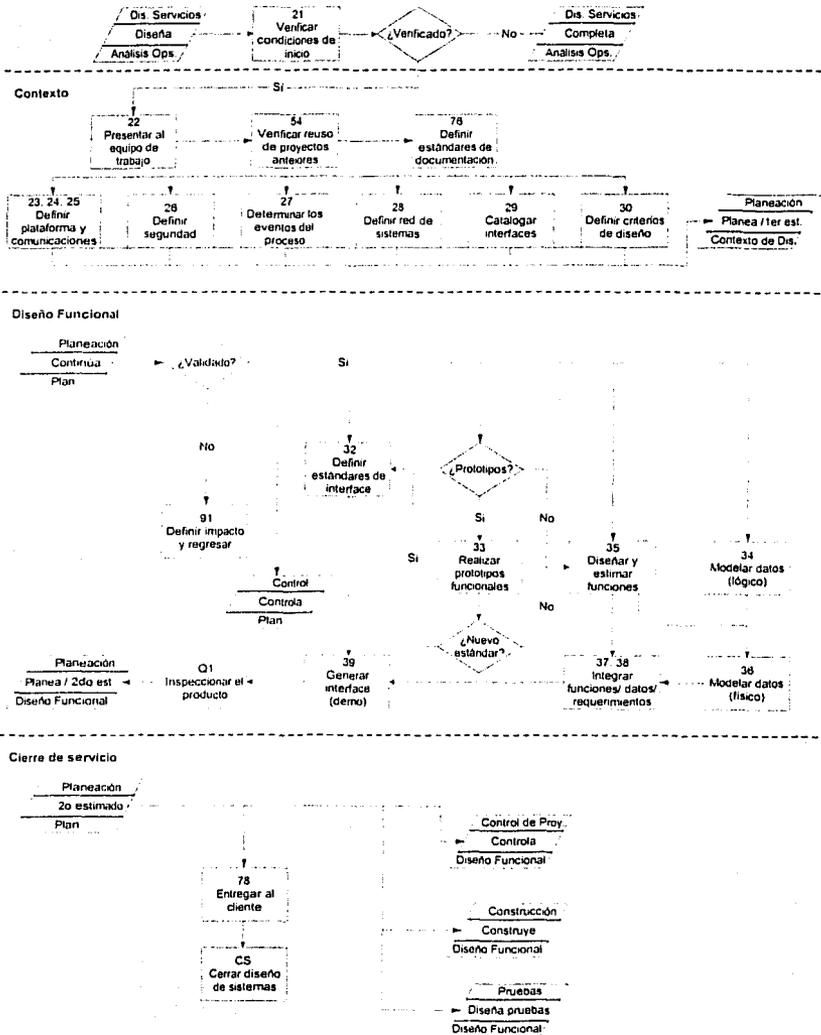


Diagrama 1.3.2 Diseño de sistemas

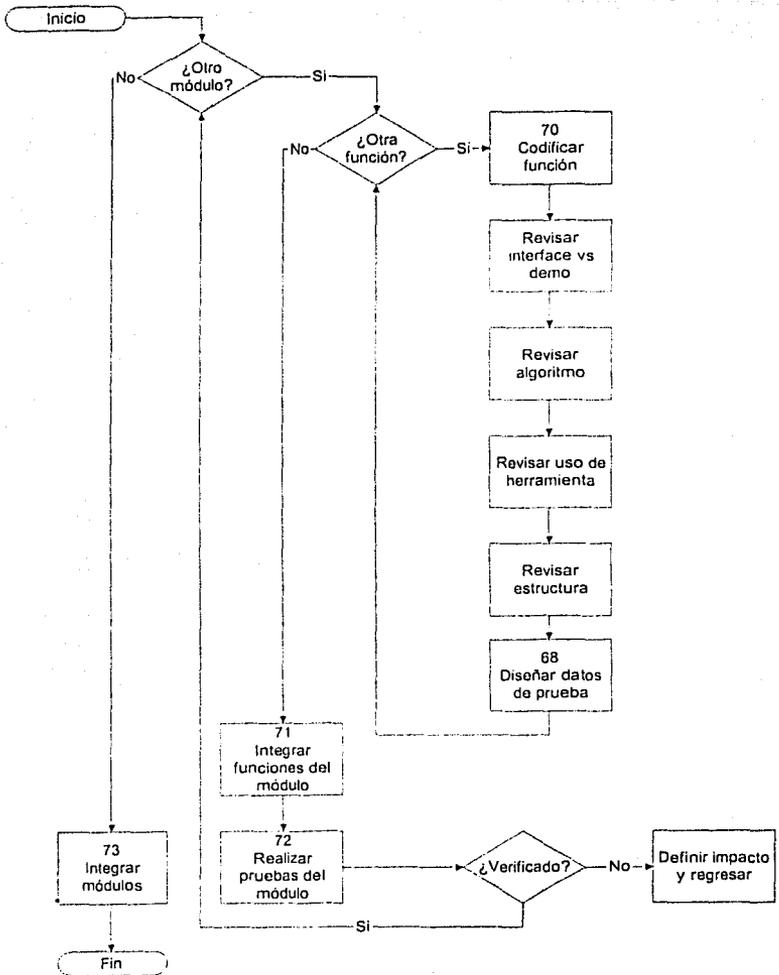


Diagrama 1.3.3 Construcción de software

1.4 El Sistema Local de Producción

Para apoyar la operación del proceso de asignación y control de proyectos, certum cuenta con una herramienta propia llamada Sistema Local de PROducción (SILPRO), cuya funcionalidad se ilustra con el diagrama 1.4.1.

"SILPRO es una herramienta diseñada y construida en SERINF, la cual permite el registro, control y seguimiento de las horas invertidas en cada una de las etapas que conforman la red de servicios."²

Por medio del SILPRO puede lograrse:

- Registro y programación de actividades.
- Control del avance y estado de cada cuenta y sus proyectos.
- Interpretación del tiempo invertido para cada uno de los proyectos.
- Comparación entre proyectos y resultados (llamados "Órdenes Múltiples – OM's) con base en estructuras estandarizadas.
- Análisis de inversiones históricas para la determinación de presupuestos para nuevos proyectos.
- Obtención de información fidedigna, oportuna, actualizada y concentrada para el costeo de operaciones.

Cuenta con los siguientes módulos:

- Infraestructura
- Catálogos de trabajo
- Seguimiento de Órdenes de Trabajo (OT's)
- Servicios
- Reportes

A continuación se expondrá el objetivo y funcionamiento de cada uno de los módulos. Se indicará al mismo tiempo a qué proceso del diagrama corresponden, indicando el nombre del proceso con mayúsculas y entre paréntesis () para su diferenciación.

² "Proceso de asignación y control de proyectos" certum, 1998

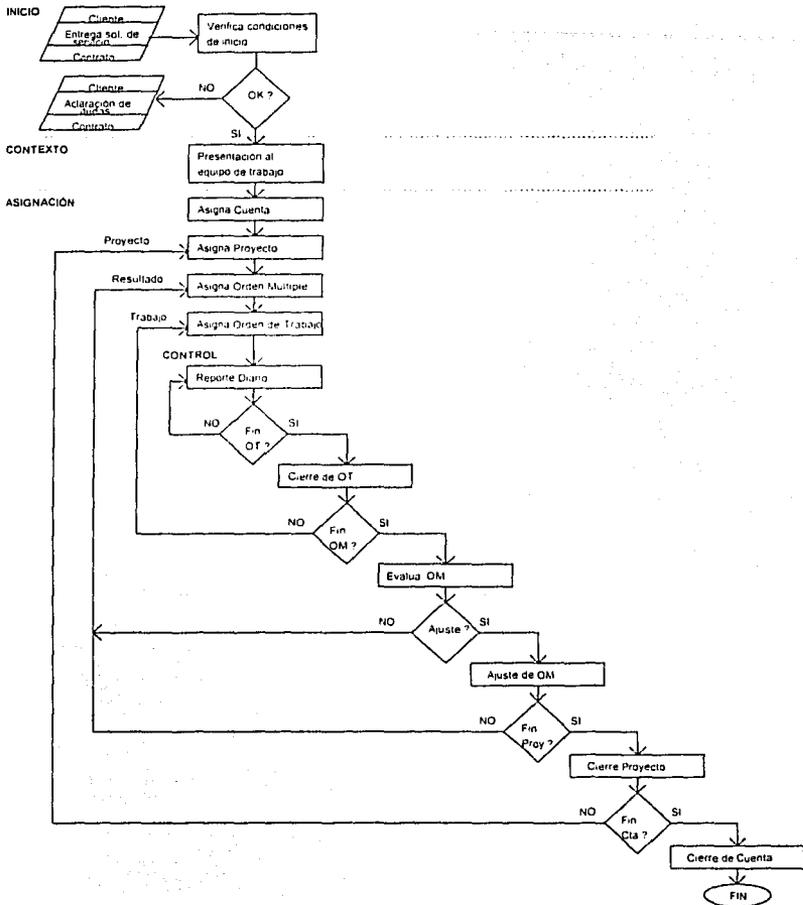


Diagrama 1.4.1 Proceso de asignación y control de funciones mediante SILPRO

1.4.1 Infraestructura

(VERIFICA CONDICIONES DE INICIO)

Este módulo tiene como objetivo establecer y mantener la infraestructura propia del SILPRO a través de los datos básicos requeridos para soportar el control y actualización de todas las cuentas. Se configuran los elementos necesarios por medio de los cuales el SILPRO pueda clasificar, agrupar y reportar la situación de los proyectos y los recursos invertidos en los mismos. Dichos elementos se clasifican en los siguientes catálogos:

1. Personas. Personas que registrarán sus actividades en el SILPRO. A cada una de las personas dadas de alta, se les asignan derechos para ingresar y/o modificar datos en el sistema sobre los módulos que le correspondan de acuerdo a sus actividades dentro de la empresa, así como una clave de acceso al sistema que corresponde generalmente a las iniciales de su nombre y apellidos, y una contraseña.
2. Tipo de OT's. En esta interfase se definen los tipos de Órdenes de Trabajo, cuya clasificación permitirá agrupar todos los trabajos realizados en la compañía como se muestra en la tabla siguiente:

Tipo de OT	Descripción
Comercialización	Conjunto de actividades destinadas a ofrecer y vender los servicios de la compañía.
Capacitación	Capacitación del personal para el diseño y desarrollo de sistemas.
Administración	Actividades relacionadas con la administración de la compañía, de las cuentas y de los proyectos.
Diseño Funcional	Trabajos relacionados a la primer etapa del desarrollo de sistemas.
Ingeniería de Detalle	Actividades relacionadas con el diseño computacional de las interfases y módulos a construir.
Construcción	Codificación de las interfaces y módulos diseñados.
Pruebas	Realización de pruebas de/entre los módulos de un sistema y entre dos o más sistemas.
Piloto	Actividades relacionadas a la instalación, piloto y monitoreo de los productos elaborados.
Diagnóstico y Corrección	Actividades inherentes al diagnóstico y corrección de problemas de funcionalidad o <i>performance</i> .

3. Secuencia default de actividades. Permite definir la secuencia de las actividades de las OT's según su tipo, la cual aparecerá por default al generarse la OT. Es un listado predefinido de *n*

actividades por cada OT.

Como la actividad representa el trabajo específico a desarrollar por una persona, y en una OT pueden participar varias personas, entonces se forma una secuencia de actividades. Por ejemplo, para una OT tipo "Construcción", pueden requerirse de las siguientes actividades:

- Diseño (ingeniería de detalle) Autor 1
- Código Autor 2
- Revisar algoritmo Autor 3
- Revisar arquitectura Autor 4
- Revisar herramienta Autor 4

4. Estados. Definen la situación de una OT y sus actividades durante su realización.

Los estados posibles para ambas son:

- Pendiente. Aparece por default al generar las OT's indicando una actividad sin concluir que puede estar en curso o pendiente de ser realizada.
- Terminada. Indica la conclusión de la actividad o de la OT.
- Cancelada. Indica el corte de un trabajo por reasignación o reprogramación. También es ocupada cuando una OT no será ejecutada.

5. Tipo de función. Define el grado de complejidad de los módulos a ser construidos, el cual puede ser:

- Básico
- Complejo
- Normal
- Simple

De acuerdo con el tipo de funcionalidad especificado puede determinarse el presupuesto para el desarrollo y venta correspondiente.

(PRESENTACIÓN AL EQUIPO DE TRABAJO)

Presentar a todos los involucrados en este proceso de Asignación y Control de Proyectos, la estructura de cuentas, sus presupuestos, sus recursos y la operación del SILPRO.

1.4.2 Catálogos de Trabajo

Módulo que permite dar de alta los distintos componentes en los que se divide un proyecto para su seguimiento y control. Los catálogos son los siguientes:

1. Cuentas (ASIGNA CUENTA)

Determinar dentro del SILPRO la Cuenta que corresponde directamente a un contrato de servicios. La cuenta es el grupo principal en la jerarquía de los catálogos. Una Cuenta es equivalente al CLIENTE (quien contrata los servicios de Diseño y Construcción de sistemas). La cuenta es el bloque principal que contiene a los demás elementos, los cuales estarán subordinados a ella.

Este catálogo solo puede ser utilizado exclusivamente por el Director de compañía.

2. Proyectos (ASIGNA PROYECTO)

Determinar dentro del SILPRO los proyectos asociados a una cuenta específica. Se determina como *proyecto*, a una subdivisión de la cuenta de acuerdo a los SISLOCS y a los proyectos de apoyo. Como parte del catálogo es necesario indicar qué persona tendrá la responsabilidad como líder del proyecto. Este catálogo solo puede ser utilizado por el director de la compañía y el Gerente de Control de Proyectos.

3. Fases

Subdivisión de un SISLOC (proyecto) de acuerdo a las fases de Diseño y Construcción de sistemas. Fases del Diseño:

- Inicio. Definir y verificar con el cliente las condiciones o requerimientos Iniciales. Se trata de reunir la información detallada de la operación que el cliente desea automatizar.
- Contexto. Todo lo requerido para soportar el diseño (plataforma, seguridad, eventos, red de sistemas, evaluar riesgos, etc.).
- Diseño. Estándares, prototipos funcionales, especificar funciones, demo de interfaces, revisión con el cliente, etc.
- Planeación y evaluación. Elaborar plan de construcción, puesta en marcha y piloto.

Fases de Construcción:

- Inicio. Verificar condiciones de inicio (recibir el diseño).
- Contexto. Verificar re-uso, definir impacto de fallas (incide en el número de revisiones) y evaluar riesgos.
- Construcción. Estándares, ingeniería de detalle, codificación y pruebas.
- Pruebas y pre-piloto. Pruebas integrales, documentación y pre-piloto (también llamado etapa de pruebas formales).

- Piloto y puesta en marcha. Implementación del producto en oficinas del cliente para últimas pruebas y arranque de operaciones utilizando el sistema.

Este catálogo solo puede ser utilizado por el director de la compañía y el gerente de control de proyectos.

4. Órdenes Múltiples (ASIGNA ORDEN MÚLTIPLE)

Determinar modularmente los resultados de cada proyecto asignado a la cuenta. Una Orden Múltiple (OM) es una subdivisión de cada fase equivalente a un resultado entregable durante las etapas de diseño y construcción de sistemas.

Se considera a una OM como una "contenedora" o "agrupadora" de OT's, ya que para lograr un resultado es necesaria la ejecución de una serie de trabajos específicos. Una OM también es un punto de control con un tiempo de entrega determinado, calculando en función del grupo de personas asignado y el presupuesto a ejercer por estas. La fecha de terminación se indica en la misma interfase. Una vez que se tengan definidos los recursos, las precedencias y los tiempos de ejecución, es posible transportarlos a la herramienta usada para el control de proyectos (detallada más adelante), para determinar la *ruta crítica* del proyecto a través de las OM's del mismo.

(ASIGNA ORDEN DE TRABAJO)

Determinar los trabajos a realizar dentro de la OM asignados a cada persona. Una Orden de Trabajo (OT) representa las actividades específicas a realizar por una o más personas en un tiempo determinado. Para subclasificar y agrupar las OT's contenidas en una OM, se utiliza el Tipo de OT (ver módulo de Infraestructura). Por ejemplo, para la construcción de un módulo, este se agrupará en tres bloques por tipo de OT: Diseño (Ing. de detalle), Código y Pruebas.

Una OT podrá dividirse en varias actividades, mas no será válido generar varias OT's para un mismo trabajo. Por lo tanto, una OT deberá contener toda la historia de un solo trabajo: secuencia de actividades, autores, horas (presupuestadas y reales) y fechas (de inicio y término por actividad).

5. Funciones. Catálogo utilizado exclusivamente para actividades de construcción.

Las funciones representan la subdivisión y especificación de la funcionalidad a codificar para los módulos que conforman un SISLOC (proyecto), por lo que antes de generar OT's de construcción es necesario haberlas definido todas.

Generalmente la función se convierte en una orden de trabajo a realizar, aunque dependiendo de la complejidad de la misma, en ocasiones es necesario segmentarla en un conjunto de programas. En esta interfase se indica el grado de complejidad de la función.

6. Programas. Catálogo utilizado exclusivamente para actividades de construcción. Representan la subdivisión de cada función en los proyectos a codificar. Siendo la mínima división de los trabajos a realizar, siempre a un programa le corresponderá una orden de trabajo.

1.4.3 Seguimiento de órdenes de trabajo

Este módulo constituye el *control* delimitado en el diagrama: Proceso de Asignación y Control de Proyectos. Su objetivo es controlar, consultar y monitorear el avance y el estado de los trabajos asignados. Está conformado por las siguientes opciones:

1. Captura de órdenes de trabajo

Interfase utilizada para la generación y asignación de Órdenes de Trabajo.

- Al generar una OT, se le asigna de manera automática, consecutiva e irreplicable, un número de folio.
- Es requerida la especificación de la Cuenta, el Proyecto, la Fase y la OM bajo la cual se desarrollará y acumularán horas de trabajo.
- Se indica la persona responsable de la OT, así como también se especifica si es cobrable al cliente o no, como pudiera ser una actividad interna.
- Para efectos de control, conocimiento y programación de los trabajos a realizar, la única persona que puede generar OT's es el líder del proyecto.
- Una vez generado el Folio de la OT, se pasa automáticamente a la interfase de seguimiento (especificada más adelante), donde se especifica por cada actividad: el autor, su presupuesto (total de horas asignadas para realizar la actividad) y la fecha prometida (fecha en que el autor se compromete a entregar la actividad terminada).

2. Seguimiento de órdenes de trabajo

Interfase de consulta, adición y modificación de las actividades que conforman cada OT. Al indicar el folio de la OT a consultar, se despliega la siguiente información:

- Ubicación de la OT en la estructura: Cuenta, Proyecto, Fase, OM.
- Tiempo total presupuestado: suma de horas presupuestadas por Actividad.
- Tiempo total invertido: suma de los registros de horas reales reportadas por Actividad.
- Estado de la OT; Pendiente, Terminada o Cancelada.

- Para cada Actividad muestra la secuencia, tipo de actividad, autor, tiempo presupuestado, tiempo real acumulado, fecha de inicio, fecha prometida, fecha del último reporte y estado de la Actividad.

La funcionalidad de esta interfase permite pasar automáticamente a la de "Captura del reporte diario", presentado a continuación, eligiendo la opción de *diario* (se indica con mayúscula la letra sobre la cual está el *acelerador* de la opción).

3. Captura del reporte diario (REPORTE DIARIO).

Registro personal de las horas ejercidas en cada una de las OT's asignadas. Este registro es necesario que sea diario, para mantener actualizada la base de datos que acumulan horas para cada cuenta. Requiere la especificación de la siguiente información:

- Fecha. Por default despliega la del día y puede modificarse para consultar o reportar algún día anterior, aunque el reporte de cada persona debe ser diario estrictamente para que la información sea fidedigna.
- OT. Indicar el número del folio (si el acceso se realizó desde la interfase de Seguimiento o de OT's pendientes por persona, este dato es heredado).
- Actividad. Indicar la Actividad a reportar (puede ser heredada también).
- Horas reportadas. Registro personal del tiempo real ejercido en la Actividad durante el día.
- Comentario. Breve descripción de lo realizado.
- Comentarios adicionales. La captura es opcional y se utilizará para detallar lo que se realizó para cumplir con la actividad. Todo lo que se indique en este espacio puede ser utilizada para análisis, pues se amplía todo con lo que se encontró el autor para realizar su actividad (problemas, soluciones aplicadas, pruebas realizadas, etc.).

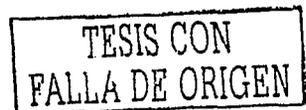
Adicionalmente se cuenta con un campo que muestra el total de horas reportadas en el día, lo cual permite mantener un control personal de los registros efectuados, e inclusive para detectar errores de captura.

4. Consulta de OT's por persona.

Permite la consulta de todas las OT's y actividades asignadas a una persona. Al seleccionar una de ellas es posible pasar al "Seguimiento de OT's" o a la "Captura del reporte diario", heredando los datos seleccionados (número de folio y actividad).

Puede consultarse a cualquier persona (pero no pueden afectarse sus registros diarios), así como monitorear sus avances y el registro diario de las horas ejercidas.

(CIERRE DE LA ORDEN DE TRABAJO)



Desde este módulo también es posible declarar terminado el trabajo asignado. La orden de trabajo en SILPRO se declara terminada al cambiarle el estado de "P" – Pendiente a "T" – Terminado. Una OT sólo podrá declararse terminada si todas sus actividades se encuentran en un estado de terminadas o canceladas.

Se han terminado de exponer los módulos que constituyen el sistema, pero no así todos los pasos para llevar a cabo el control del proyecto. Los pasos restantes se indican a continuación:

(EVALUACIÓN DE LA ORDEN MÚLTIPLE)

El objetivo de este paso es revisar, analizar y evaluar los resultados alcanzados en la Orden Múltiple, además de declararla terminada o cerrada mediante un cambio de estado una vez que todas sus órdenes trabajo se encuentren terminadas o canceladas.

La evaluación consiste de un análisis de diferencias entre horas presupuestadas a cada actividad y las horas reales reportadas que se invirtieron en ellas. Dicho análisis puede arrojar ahorros de horas (presupuestadas > reales), o bien, excesos (presupuestadas < reales). Los resultados son utilizados para evaluar tanto los resultados totales del proyecto como a cada persona involucrada en el mismo.

(AJUSTES PARA ÓRDENES MÚLTIPLES POSTERIORES)

Su objetivo es aprovechar los resultados obtenidos de la evaluación de la OM anterior par aplicarlos y realizar ajustes a OM's posteriores. Los ajustes deberán realizarse a OM's con resultados esperados análogos a los evaluados en la OM anterior.

(CIERRE DEL PROYECTO)

Declarar terminado o cerrado el proyecto en el SILPRO implica el cierre de todas sus OM's y la apertura de una OM llamada "de Garantía". El proyecto permanecerá en estado "P",- Pendiente hasta que se cumpla el periodo de garantía y se cierre la OM correspondiente. El periodo de garantía se fija por contrato y es por lo regular de 1 año.

(CIERRE DE LA CUENTA)

Una cuenta sólo podrá declararse terminada o cerrada si todos sus proyectos lo están. Después de que una cuenta se encuentra en estado "C" – Cerrada, podrá utilizarse la opción que permite pasar a archivos históricos dicha cuenta.

1.4.4 Servicios



Este módulo tiene como objetivo permitir la realización de cambios, ajustes o re-estructuraciones de los catálogos y OT's existentes en el SILPRO. Está compuesto de los siguientes servicios:

1. Cambio de OM para OT.

Es utilizado para la reubicación de una OT, ya que permite el traspaso de la misma a cualquier otro lugar en la estructura del SILPRO. Es utilizado generalmente para la actualización de nuevos criterios en la estructuración y manejo de la Cuenta, o para corregir errores en la generación de OT's. Para lograr el traspaso es necesario indicar a qué Cuenta, Proyecto, Fase y OM respectivamente se ubicará la OT.

2. Cambio de Fases para OM.

Esta funcionalidad es utilizada para re-estructuraciones de Órdenes múltiples en una Cuenta y actualización de criterios para el manejo de la misma. A diferencia del servicio anterior, esta interfase permite realizar el traspaso de una OM completa (con todas las OT's que ésta contenga) a cualquier otro lugar en la estructura del SILPRO. Para lograr el traspaso es necesario indicar a qué Cuenta, Proyecto y Fase respectivamente en las que se ubicará la OM.

3. Cambio de Generales de una OT.

Permite modificar la información básica de una OT. Se utiliza para corregir un registro erróneo, o cuando ha cambiado algún criterio o el responsable de la OT. Una vez indicado el número de la OT, despliega los datos existentes para ella, permitiendo la modificación de:

- Descripción de la OT
- Cobrable (S/N)
- Función
- Programa
- Responsable

4. Pasar a Histórico una Cuenta cerrada.

A través de esta interfase es posible obtener el respaldo de un Proyecto declarado como Terminado, convirtiéndose este en un "histórico" de la Cuenta.

1.4.5 Reportes

Este último módulo genera la información requerida para el Control y Seguimiento de las Cuentas y Proyectos existentes en el SILPRO, presentando los registros de OT's y horas reportadas por cada persona. Contiene los siguientes reportes:

1. OT's Pendientes por persona
2. Seguimiento de Órdenes Múltiples
3. Resumen Cuenta/Proyecto por persona
4. Genera ASCII para Admón.

Genera un reporte *ASCII* de las horas presupuesto-reales por Cuenta-Proyecto-Persona durante un periodo determinado, subclasificándolas en cobrables y no cobrables. Se utiliza para obtener un resumen de los trabajos realizados en SERINF durante un mes por Cuenta-Proyecto, desglosando la participación de cada persona en cada uno de los diferentes proyectos. Una vez obtenido puede ser importado a una hoja de cálculo para realizar el proceso de Costeo de Operaciones, detallado más adelante.

5. Resumen Cuenta/Proyecto de hrs. Excedidas
6. Seguimiento de Proyecto por Actividad
7. Reporte Diario de horas Cuenta/Proyecto
8. Reporte Diario de horas por día
9. Reporte de Registro Diario
10. Resumen de Cuenta
11. Resumen de Cuenta con tipo de OT
12. Historial de Orden Múltiple

1.5 La Planeación y Control de Proyectos

En el diagrama de "Red de Servicios" presentado anteriormente, es visible la estrecha relación que guardan los distintos servicios ofrecidos con la planeación y control, esperando que sean de utilidad para lograr los máximos rendimientos y los mayores beneficios a los clientes. Las políticas para guiar el control del proyecto pueden ser adecuadas formalmente a las necesidades específicas de cada cliente o de cada proyecto, pero sin omitir las tareas que aseguran la calidad del producto en confiabilidad y tiempo de entrega (puntos de control, o denominados también "milestones oficiales").

Con el adecuado control de los proyectos se pretende cubrir tres funciones básicas:³

1. Inspección

- Asegurar que los usuarios se basen en el machote de MS-Project que modela los procesos estándar de cada uno de los servicios certum, pues es la base para generar el Plan del Proyecto. Este machote diseñado por certum, lleva el nombre de ©Estliga.
- Revisar que todo sea presupuestado y que el detalle corresponda al estimado en curso, y no solo a lo que le interesa al responsable del servicio en turno; por ejemplo, planear la construcción con el estimado total olvidando las inspecciones y pruebas formales.
- Revisar que se registren tareas extrañas al proceso pero que afectan el desarrollo.
- Revisar que se registren las tareas para cobrar al detalle necesario: hacer la estimación, llevar a cabo los trámites de cobranza, depositar en el banco, etc.
- Que después de inspeccionar el estado actual, no quede ninguna tarea con fecha de inicio vencida o de terminación vencida y con avance 0%; actualizar el Plan de acuerdo a actividades reales para impactar las tareas subsiguientes.

2. Control y Coordinación

- Advertir a los responsables de otros servicios del proyecto, o de otros proyectos, sobre el impacto por las reprogramaciones de otros.
- Coordinar la ocupación del personal entre las diferentes Cuentas.
- Intervenir y coordinar acciones urgentes entre proyectos.
- Analizar y llamar a la reflexión sobre datos macros como: horas remanentes y resultados de la comparación de presupuestos (p.e. pruebas de integración vs. pruebas formales, desviación de la nueva estimación vs. la esperada).

³ "Guías para la planeación y control de proyectos", certum 1994

3. Capacitación y mejoramiento continuo.

- Capacitar y apoyar al personal que lo requiera.
- Actualización constante sobre mejoras en el uso de Project (riesgos, *bugs* descubiertos y sugerencias operativas) así como en las técnicas mismas de Planeación y Control.

Con la gráfica obtenida de la captura de los datos solicitados por el ©Estliga, se espera:

1. Modelar el proceso de cada proyecto en particular.
2. Controlar las fechas de inicio y de fin.
3. Comparar los presupuestos de trabajo contra el trabajo real.
4. Identificar y aislar las *tareas críticas* y los puntos de control.
5. Administrar la capacidad requerida de recursos tipo asignados.

Los recursos asignados a las tareas en el ©Estliga son:

1. Ingeniero 1: hace diseños y programas simples.
2. Ingeniero 2: hace diseños y programas complejos.
3. Ingeniero 3: Similar al Ingeniero 2 pero además puede controlar varios ingenieros y sabe hacer diseños funcionales e ingeniería de detalle. Tiene a su cargo uno o varios módulos de un sistema.
4. Ingeniero 4: Similar al Ingeniero 3 pero además sabe hacer análisis de operaciones. Es responsable de la integración de todos los módulos de un sistema.

Una adecuada planeación debe ofrecer áreas de oportunidad como:

1. Administrar productivamente toda la capacidad del personal a cargo de las tareas no críticas (que por lo general son la mayoría), esto es, ocupar el tiempo libre si es que lo hubiere, en inversiones o capacitación.
2. Archivar como prototipos los planes calificados como exitosos o como erráticos, con información de referencia. Esto ayudará en futuros presupuestos de proyectos.
3. Llegar a tener la información y la técnica para utilizar el método PERT, si se llegara a justificar.

1.5.1 Estructuración del Proyecto

TIPOS DE PROYECTOS:

1. Nuevos sistemas: Su estructura deberá basarse en el ©Estliga.
2. De mantenimiento: Son nuevos requerimientos del cliente; su estructura está en el SILPRO.
3. De garantía: Son defectos reclamados por el cliente.

ESTRUCTURA DE PLANES (GANTT):

1. El plan global de la cuenta ("master") agrupará todos sus proyectos específicos según la clasificación acordada con el cliente, p.e. nuevos desarrollos de la fase anterior, de la fase actual, de la fase futura, de investigación, de mantenimiento, etc.
2. El *master* debe estar estructurado de tal modo, que el cliente pueda ver todo en forma claramente acotada, agrupada y ordenada los grupos de proyectos que lo componen, de modo que pueda aislar su atención en éstos y verlos íntegramente (no disperso) y, que sirva de base para una discusión o una renegociación.
3. Cada proyecto debe corresponder a un contrato con el cliente, esto es, que si solo se ha contratado un servicio (p.e. diseño del sistema) se hará un plan acotado solo a esto y se dará de alta a este proyecto en el SILPRO. Los siguientes servicios que se contraten serán separados de la misma forma en su plan y en SILPRO.
4. El plan de un proceso específico de un proyecto será único, por esto es permitido hacer todas las adecuaciones necesarias a la copia del ©Estliga, como borrar OM's, OT's, modificar actividades dentro de una OT, modificar procedencias entre tareas y entre actividades. Pero s no está plenamente justificado, está prohibido borrar los puntos de control de calidad oficiales precisados desde el ©Estliga.

Se recomienda el manejo de distintos colores en las OT's para diferenciarlas al elaborar y analizar el ©Estliga:

Rojo: para control de calidad, de estimados y puntos de control.

Azul: para tareas de externos y tareas para la cobranza.

Verde oscuro: para tareas de programación de funciones.

Magenta: para los amortiguadores de atrasos y estimados de todo el servicio y/o fases.

PUNTOS DE CONTROL OFICIALES

En ©Estliga están marcados como puntos de control oficiales:

1. Las tres inspecciones formales
2. Las definiciones de procesos
3. Las dos revisiones finales con el cliente
4. La elaboración del plan del 1º, 2º y 3er. Estimado
5. Las pruebas del Sistema

6. Las pruebas integrales de Sistemas
7. Las pruebas de aceptación
8. Entrega final al cliente
9. Las tareas de amortiguación de retrasos de cada servicio

El responsable del proyecto puede insertar estratégicamente puntos de control para controlar su proyecto. También pueden ser por acuerdo con el cliente o con el responsable de producto.

1.5.2 Control de incidentes

Se conoce como incidente a "todo defecto en diseño o construcción, cambio de especificación o nuevo requerimiento del cliente, descubierto y registrado durante el desarrollo, el piloto o la operación del sistema"⁴. Son registrados y administrados en los repositorios de cada proyecto (en Lotus Notes). Se diferencian y se atienden según su severidad y programación.

⁴ Base de datos de incidentes de certum

SEVERIDAD

- Los incidentes numéricos (00 a 03) son defectos imputables a certum.
- Los incidentes alfabéticos (A a D) son cambios de especificación o nuevos requerimientos, del cliente.

PROGRAMACIÓN

Se refiere al momento en el que el incidente será asignado a algún ingeniero para su solución.

- Inmediata e ininterrumpida.
- Siguiete entrega.
- Siguiete versión.
- Por definir. Cuando una persona distinta del responsable del módulo levanta el incidente, debe indicarle como programación "Por definir", en espera de que el responsable le asigne la programación que le corresponde.

SEVERIDAD VS. IMPACTO OPERATIVO Y PROGRAMACIÓN

- Severidades 00 y A: Detiene la operación del cliente y/o hay daños severos. Son de urgencia porque se detuvo la operación o las pruebas del cliente, total o parcialmente; también puede ser porque se están alterando o destruyendo datos y/o procesos. Acción: inmediata e ininterrumpida hasta reestablecer la operación sin reprogramación.
- Severidades 01 y B: Son programadas, porque aún con el incidente la operación continúa o ya se instaló una solución provisional. Acciones: Replanear para incorporarlos en la siguiente entrega.
- Severidades 02 y C: Son programados para la siguiente versión, por acuerdo con el cliente.
- Las severidades 03 y D: Se relegan, son incidentes que no afectan la operación (p.e. ajustar posición de botones) o son deseos de mejora a futuro, que pueden pasarse a la siguiente versión. Acciones: se revisan periódicamente con el cliente a fin de asignarle prioridades en el plan de trabajo de la cuenta o cancelarlos definitivamente.

En el diagrama 1.5.2.1 se ilustran los diferentes estados por los que pasa un incidente desde que es detectado hasta que se considera Cerrado.

PROCESO DE RESOLUCIÓN

Para cambiar el estado de un incidente de Autorizado a Resuelto, la persona asignada debe cumplir con los siguientes puntos:

- Diagnosticar la causa del incidente y dar un presupuesto para su solución.
- Determinar el plan de solución señalando los cambios necesarios para corregir el error o satisfacer el requerimiento tomando en cuenta el impacto que pueda provocar.
- Llevar a cabo el plan de solución. Si se requiere, obtener los archivos a modificar del controlador de versiones.
- Realizar las pruebas necesarias comprobando que el error ya no se presenta o que el requerimiento se ha cubierto.
- Crear una nueva versión de los archivos modificados en el controlador de versiones indicando los cambios realizados y el folio del incidente que provocó tales cambios.
- Incluir en el incidente la ruta completa de los archivos modificados y la versión de los mismos.
- Anexar, si aplica, las peticiones de cambios en base de datos necesarias.

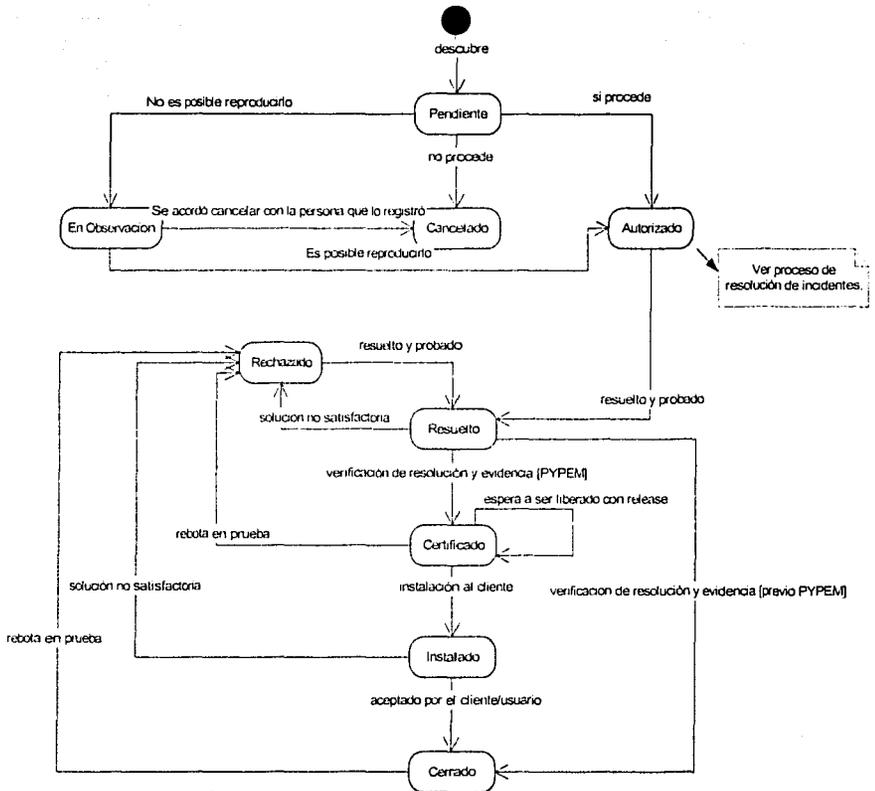


Diagrama 1.5.2.1 Diagrama de estados de incidentes

1.5.3 Mantenimiento del plan

A continuación se presentan los puntos que señalan cómo llevar a cabo el plan de los proyectos y las tareas a llevar a cabo para su correcta y oportuna actualización de acuerdo al avance de los mismos.

El responsable del proyecto es la única persona destinada a elaborar el plan con base en el @Estliga, modificarlo, actualizarlo y custodiarlo, así como asegurarse de que su consulta esté disponible para todo el equipo de trabajo para que se mantengan informados del estado del proyecto. Dicha consulta puede estar al alcance de manera electrónica mediante una copia del archivo de "sólo lectura", o bien, de manera impresa y colocada en la pizarra propia del proyecto. La existencia de la versión impresa es obligatoria y es conocida como "panel de control".

Así mismo, el responsable del proyecto se hará cargo de supervisar el avance de sus proyectos y de actualizar su *master* correspondiente, si es que lo hay. El *master* se necesitará cuando los proyectos en desarrollo del cliente sean tantos, que se requiera de una vista integral y estructurada del resumen de todos ellos para revisar el conjunto.

TAREAS DE INCIDENTES

Las tareas de incidentes autorizados de cualquier proyecto, no se programarán en su plan correspondiente sino que se controlarán en el SILPRO, asignándolas al proyecto que correspondan. Se programan planes especiales de incidentes solo si la situación es crítica y se requiere de una fase extraordinaria de correcciones y pruebas.

COORDINACIÓN DE ACTUALIZACIONES

Para llevar a cabo la coordinación de actualizaciones, el Ejecutivo de la Cuenta debe tener, el día y la hora convenida, los Gantt's de todos sus proyectos, actualizados al día anterior, esto es:

1. Con folios de OT's abiertas en SILPRO, registradas en el Gantt.
2. Replaneados y/o reprogramados.
3. Con avances del 100% de las tareas con fechas de fin vencidas.

4. Siempre se actualizará el día/mes al inicio del nombre del 1er. Renglón del Gantt, para hacer visible a qué día está actualizado, p.e. "22/Ago Operar Compras y Ofertas".

PRESENTACIÓN DEL GANTT AL CLIENTE

Para evitar falsas expectativas del cliente y fijación de fechas de fin, el Gantt se presentará al nivel de detalle que corresponde al estimado; esto es, presentando las columnas de datos y las escalas del diagrama de barras correspondientes:

Durante el estimado 1:

- Nombre del servicio.
- Columna de fecha de inicio y de porcentaje completado.
- Escala mayor: mes.
- Escala menor: nada.

Durante el estimado 2:

- Nombre del servicio y sus fases (opcional).
- Columna de fecha de inicio y de porcentaje completado.
- Escala mayor: mes.
- Escala menor: semanas (porque no hay quincena en Project).

A partir del estimado 3:

- Nombre del servicio y sus fases.
- Columna de fecha de inicio y de término, y de porcentaje completado.
- Escala mayor: semana.
- Escala menor: día.

Nota: Por requerimiento del cliente, es posible que se presenten más datos.

ENFOQUES PREVENTIVOS PARA CONTROL

TESIS CON
FALLA DE ORIGEN

Además de supervisar lo usual, esto es, inicio y terminación de tareas, porcentaje de avance, retrasos, fecha final de cierre del proyecto comportamiento de la *ruta critica*, es necesario cuidar lo siguiente:

1. ¿Quiénes son las personas asignadas a las tareas de la ruta crítica? Debe ser personal que por su conocimiento y seguridad garanticen que no haya atrasos imputables a la responsabilidad de la persona.
2. Calcular las horas/día remanentes al arrancar el proyecto y cuando se considere conveniente (quincenalmente, mensualmente o cuando ocurra un cambio fuerte), esto servirá para evaluar si la capacidad asignada puede realizar los trabajos, p.e. se tienen 100 hrs/día remanentes y hay 5 personas asignadas; entonces, esto indica que van a trabajar 20 hrs/día, lo que implica que se hizo mal la asignación de capacidad y no se va a cumplir.
3. Uso de las tareas amortiguadoras de cada servicio:
 - Presupuestarles horas de trabajo.
 - Reducir su duración en la misma medida del atraso de la fecha final.
 - No se amortigua por nuevos requerimientos, se replaneará.

COMPRESIÓN

Partiendo de que los presupuestos de trabajo están bien calculados, para acercar la fecha de finalización del proyecto, por estimación o por atraso:

1. Restar duración al amortiguador(es) de atrasos previsto(s).
2. Reprogramar. Implica trabajar más horas por día por persona, trabajar en días inhábiles y asignar más personal, si la hay y conviene. Contemplar si conviene, pues es probable que más que ayudar, implique más horas de inducción al proyecto e incluso a la herramienta.

LISTAS DE VERIFICACIÓN (checklists)

Se cuenta con listas que contienen actividades que debieron haber sido ejecutadas durante la planeación del proyecto. Su objetivo es ayudar al planeador para que elimine errores, principalmente por olvido, y que reflexione sobre temas clave. Su uso es recomendable.

TESIS CON
FALLA DE ORIGEN

Conclusiones

Como se puede apreciar *certum*, la fábrica de software sobre cuyo estudio está basado este trabajo, ha tenido presente desde sus inicios los principales conceptos de planeación en las empresas, entre los que se encuentran: la definición de misión, visión (apartado 1.2), servicios, políticas (1.3) y valores (los valores de la empresa se enumeran más adelante, en la sección 2.1.2).

Ofrece a sus clientes los servicios de diseño, construcción y pruebas formales de aplicaciones a la medida; la planeación de cada uno de ellos, así como de todo el conjunto, se define en especificaciones debidamente documentadas siguiendo las normas de calidad que le adjudicaron la certificación ISO 9001. Se genera así el ©Diseño de Servicios, conjunto de procesos y reglas sobre el cual se rigen los productos finales. La organización en general y los resultados obtenidos hicieron posible también la obtención del nivel 3.76 de productividad, según el estándar internacional SPICE.

Cuenta también con procedimientos, detallados en los diagramas de diseño y construcción de sistemas que se encuentran en las pags. 14 y 15 respectivamente, que ilustran los pasos a seguir para llevar a cabo un plan de desarrollo para cada uno de los servicios. La sección 1.4 expone la base fundamental de la planeación en *certum*, la herramienta SILPRO (Sistema Local de Producción), misma que ofrece una manera eficaz y sencilla de almacenar, y de esta manera controlar, todas las actividades ejercidas por cada uno de los ingenieros que participan en la realización de un producto. Su característica más importante es el reporte diario de horas/hombre, pues a partir de esta información junto con el plan formal denominado ©Estliga (archivo en Microsoft Project) se hace posible la estimación aproximada de costos, la delimitación de puntos de control y tiempos de reserva y el análisis de rendimiento del personal, entre otras cosas.

El desmesurado avance tecnológico obliga el mantenimiento constante de las técnicas no solo de planeación, sino en general de los procesos que tienen que ver con el desarrollo de sistemas, así como de la conjunción entre ambos. Se realizará entonces un análisis de las áreas mencionadas para poder determinar los puntos que habrán de afinarse para poder garantizar el continuo cumplimiento de la política de calidad: calidad, costo, fecha de entrega.

FALTA

PAGINA

39

CAPÍTULO II

Análisis de las técnicas utilizadas para planear y controlar los proyectos

Objetivo: *Se realizará un análisis de la manera como actualmente se planean y controlan los proyectos, realizando una investigación de las técnicas utilizadas, el por qué de su implementación en su momento y la efectividad que han reflejado en su intento por reducir la incertidumbre en la confiabilidad de los productos terminados, la fecha de su liberación y la diferencia entre el costo real generado por su desarrollo y el pronosticado al inicio del proyecto.*

2.1 Análisis de la planeación en SERINF

SERINF se encuentra en una etapa de crecimiento originada por requerimientos de construcción de aplicaciones en Internet. El crecimiento es visible por el nuevo personal que se integra y que origina la implementación de más infraestructura (computadoras, cableado de red, extensiones telefónicas, mobiliario, etc.), así como la necesaria adquisición del software utilizado y documentación del mismo.

Tomando en cuenta la diferencia entre *crecimiento* y *desarrollo*¹ de una empresa, es claro que el desarrollo no se dará automáticamente por estos nuevos requerimientos, sino que es necesario un estudio de los procesos de producción actualmente utilizados, que repercuta en una actualización de los mismos para asegurar la posición hasta ahora lograda en el mercado por el desarrollo de sistemas y además, incursionar en el de aplicaciones en Internet. Este estudio de procesos se traduce al análisis de la planeación de proyectos trazada por SERINF así como del seguimiento que se le da a la misma a lo largo del desarrollo de los proyectos.

2.1.1 Tipo de Planeación

Siguiendo la tipología de la planeación presentada por Stephen P. Robbins², a continuación se presenta un análisis de la orientación tomada por SERINF para realizar la planeación de sus proyectos.

Desde su inicio de operaciones, la empresa ha tratado de captar el interés de clientes que se distinguan por contar con volúmenes y tipos de operación cuya automatización represente un reto. Esta determinación persigue dos fines: el primero considerado como *objetivo externo*, porque obedece al crecimiento y desarrollo de la empresa asegurando así el logro de una buena posición en el mercado donde se encuentra situada, y el segundo que puede ser determinado como un *objetivo interno*, pues se refiere a lineamientos que SERINF se ha establecido para el impulso y motivación de su personal.

El objetivo externo se refiere a obtener contratos cuyos proyectos cuenten con un cierto grado de dificultad de tal manera que estén lejos del alcance de una parte de la competencia, es decir, de aquéllas

¹ ACKOFF, Russel L "Planificación de la empresa del futuro" Ed. Limusa, México pag. 52-55

² ROBBINS Stephen P. "Administración, teoría y práctica" Ed. Prentice Hall Hispanoamericana, México 1987

compañías que prefieren comprometerse a la realización de sistemas parecidos a los que ya se han tratado anteriormente y que garantizan de alguna forma el éxito al ser liberados. Se espera que el cliente sienta cierta confianza al notar que la empresa está decidida a aceptar el reto de construir sistemas a cuyos procesos, interfaces y tecnología requerida, no cualquier otra empresa estaría dispuesta a comprometerse. Por supuesto que esto crea una buena imagen en el mercado, pero no sólo se trata de asumir la responsabilidad, sino de cumplir con ella en todos los aspectos. Por esta razón, los ingenieros³ son elegidos con sumo cuidado, aplicando a cada aspirante exámenes de análisis de algoritmos, de personalidad y de manejo de la herramienta para la cual se busca contratar personal. Es claro entonces, que la empresa debe cuidar de un aspecto muy importante: proporcionar a los ingenieros con los que cuenta, proyectos cuya realización logre mantenerlos actualizados en el negocio del desarrollo de software. Este punto representa el objetivo interno que se mencionó anteriormente.

Continuando con el análisis de los objetivos de certum, encontramos que sus objetivos externos pueden enumerarse de la siguiente manera:

- Asegurar la máxima prosperidad tanto para los inversionistas como para las personas que laboran en la empresa.
- La permanencia de la empresa en el mercado mediante el cumplimiento de los requerimientos del producto (calidad), del precio y del servicio, siguiendo ese estricto orden.

Se catalogan como *externos* porque no son los objetivos originales de la creación, estructuración y mantenimiento de la empresa, sino que son lineamientos internos que han sido propuestos como *medios* para alcanzar los objetivos externos y cuyo cumplimiento no solo cumple el requerimiento de obtener utilidades sino el hecho de saber que las cosas se están haciendo bien y esto resulta en satisfacciones tanto para los accionistas y personas que ocupan los puestos de dirección, sino también para todos los ingenieros, pues los tres colaboran en su desarrollo profesional e incluso personal.

Para citar los objetivos internos, podemos acudir a lo que dicta la Misión de la empresa:

"Garantizar la implementación exitosa de servicios estratégicos que hacen uso creativo de tecnología de información, con la menor inversión"

Analizando entonces los requerimientos a los que responde certum, podemos relacionarlos con los que cita Antonio Saldivar como requerimientos técnicos y sociales⁴, donde los técnicos se refieren a aquéllos

³ En certum se le denomina *ingeniero* a toda persona que colabore en el desarrollo de cualquiera de los servicios ofrecidos. Se excluye entonces al personal administrativo y de soporte interno

⁴ SALDIVAR, Antonio. "Planeación financiera de la empresa" Ed. Trillas pag. 11



que se relacionan con las actividades productivas y constituyen su objetivo básico o formal. En este caso, se trata del desarrollo de tecnología de información. Los sociales van ligados con las expectativas de las personas que intervienen directamente o indirectamente en su operación. Estas personas se incluyen implícitamente en la misión de certum, pues "el uso creativo de tecnología de información" es llevado a cabo por los ingenieros de producto, quienes continuamente deben actualizar sus conocimientos para poder manejar e implementar tecnología de punta. Por el otro lado, los clientes siempre buscarán la mejor calidad al menor precio, es por eso que se les considera en la misión, brindándoles la oportunidad de contar con un buen producto "con la menor inversión". El responder a estos dos tipos de requerimientos define a la empresa como un *sistema tecnosocial*⁵, esto es, un conjunto de recursos y elementos económicos, materiales, técnicos y humanos que actúan en forma coordinada para producir determinados bienes y servicios.

Hasta ahora se han expuesto y desglosado algunos de los puntos más importantes de la manera de trabajar de certum, y es suficiente para afirmar que los proyectos se desarrollan siguiendo una planeación de tipo "formal". A continuación se presentan las características con las que debe contar este tipo de planeación, algunas de las cuales ya se ha expuesto la forma en la que son definidas por certum; las que aún no se han especificado en este análisis se desarrollan ahora. Cabe mencionar la diferencia que existe entre la planeación *formal* y la *informal*, y es, que la última "se basa únicamente en la experiencia previa de las personas que llevan a cabo la planeación después de haber participado en la ejecución misma por lo menos como observadores"⁶. Cuando la planeación es informal no hay nada escrito, basta confiar en lo que dictan los directores.

Elementos de la planeación formal:

1. Objetivos y medios.

Se han expuesto los objetivos. También los medios aunque quizás no explícitamente. Uno de los medios más importantes es el reporte diario de actividades mediante la herramienta SILPRO, cuyo funcionamiento y beneficios fueron desglosados en el primer apartado del presente trabajo. Otro medio es la existencia de una *base de datos* ubicada físicamente en uno de los *servidores* con los que cuenta la infraestructura de la *red local* implementada en la empresa. Pero el medio del que se habla, no existe solamente por la presencia de la base de datos, sino que ésta, que lleva como nombre "Servicios certum", está disponible para su consulta por parte de todos y cada uno de los miembros que componen el personal. La accesibilidad de la información que se guarda en dicha

⁵ Término acuñado por Emery y Trist, del Tavistock Institute de Londres. "Socio Technical Systems", incluido en "Systems Thinking", Penguin, 1969

⁶ ROBBINS Stephen P. "Administración, teoría y práctica" Ed. Prentice Hall Hispanoamericana, México 1987

base de datos es importante porque permite que toda persona que forme parte de la empresa, trabaje siguiendo los mismos métodos, asegurando así la homogeneidad de los servicios prestados a cada una de las cuentas (clientes) que son atendidas. "Servicios certum" contiene toda la documentación sobre historia, visión, misión, organigrama y políticas, así como los procesos, planes y estrategias a seguir para llevar a cabo el Diseño de Servicios©. Toda la información puede ser accesada buscando, ya sea por la clasificación del documento, o bien, por la norma de calidad a la que obedece.

Así, los medios para alcanzar los objetivos externos e internos, están determinados por lo que dictan los procesos del Diseño de Servicios© ubicados en la base de datos "Servicios certum" y la disponibilidad de los mismos para su consulta.

2. Misión (de la empresa)

"Garantizar la implementación exitosa de servicios estratégicos que hacen uso creativo de tecnología de información, con la menor inversión"

3. Visión (de la empresa)

"Ser la mejor empresa de proyectos de ingeniería de software en México para los mejores clientes y los mejores ingenieros."

4. Diagnóstico. Emitido en su mayor parte basado en los resultados de los distintos tipos de análisis procesados por SILPRO a través de los archivos que contienen la historia de las actividades realizadas para el desarrollo de los productos, incluyendo el tiempo invertido en ellas, y así, el costo generado.

5. Pronóstico. A partir del diagnóstico, que otorga un análisis situacional de los proyectos, y así, de la planeación de los mismos, es posible emitir un pronóstico de cómo se comportarán los proyectos subsecuentes; específicamente, del costo que se deriva de su construcción, y de esta manera, de las posibles utilidades.

6. Políticas. En la base de datos Servicios certum, se establecen como políticas generales:

- Circuito de calidad. Consiste de un diagrama que refleja el compromiso con al calidad y satisfacción del cliente.
- Control de documentos. Especifica las medidas de seguridad y control de creación y actualización de documentos internos (para uso del personal de la empresa), externos (para entregar a clientes, como manuales, guías, normas, etc.) y legales.
- Contratos. Formalización legal de los servicios pactados con el cliente.

- Sistema de calidad. Esta política cumple con los requerimientos de la norma ISO-9001 documentando el sistema de calidad de certum. Está organizado de acuerdo a los servicios, procesos, procedimientos y actividades que se siguen en la empresa e incluye todos los servicios de la red de servicios certum.
 - Revisiones de la dirección. La descripción de esta política es: "Cada doce meses, el representante de la dirección presentará los resultados de la revisión del sistema de calidad a la dirección para evaluar su efectividad y si satisface los requerimientos de nuestras políticas de calidad, así como las de las normas internacionales que nos rigen. Las no conformidades se registran en la BD Auditorías y las acciones correctivas que se tomen actualizarán la BD Servicios certum."
 - Política de calidad. Dicta que:
 - i. certum y empresas asociadas tienen como objetivo mantenerse como líderes en la Confiabilidad, Tiempo de Entrega y Costo (en ese orden) de los servicios que ofrecen.
 - ii. Todo el personal que presta sus servicios en estas empresas tienen obligación directa por la Confiabilidad, Tiempo de Entrega y Costo dentro de sus áreas (proyecto, servicio) y niveles de responsabilidad.
 - iii. La estrategia para la realización de estos objetivos y política se llevan a cabo solamente a través de la medición y mejora de las 4 Ps operativas: Proceso, Personas, Producto y Proyecto.
7. Programas. Tanto de los servicios de diseño y construcción como de planeación y control (Incluidos como anexos). Salta a la vista que hace falta un programa general que proponga un plan de acción a seguir para llevar a cabo las pruebas de los productos. Hasta el momento se encuentra estipulado que cada ingeniero es responsable del producto entregable que se deriva de cada tarea que tenga asignada. Esta responsabilidad obliga a que él mismo aplique las pruebas que considere necesarias para aseverar la calidad de sus entregables, aunque esté presente que el producto final debe pasar por un periodo de pruebas formales, las cuales se encargan de realizar sobre los sistemas, las operaciones para las que fueron creados. Las operaciones deben ser ejecutadas en su totalidad, pasando por todos los procesos internos y transacciones, tal y como serían realizadas por el usuario final e incluso siguiendo secuencias anormales sobre el sistema para verificar que todos los errores que puedan ser provocados por el usuario, estén contemplados y que no ocasionen errores de sistema como una inesperada terminación del mismo, o que los resultados arrojados no sean los esperados. Por ejemplo, si el usuario se encuentra dando de alta los datos de una persona en el sistema, el resto de los botones cuya acción está destinada a modificar o eliminar a la persona,

deben presentarse deshabilitados mientras no se dé por terminada la acción de *crear a la persona*, pues de otra manera se generaría un error al tratar de modificar o eliminar datos que en realidad aún no existen (porque no se han guardado en la base de datos). Este tipo de situaciones están contempladas en documentos que el ingeniero debe tomar en cuenta, ya sea durante el desarrollo de funciones, o bien, al aplicar las pruebas a su producto entregable. Pero dichos documentos no son generales para su aplicación a todos los proyectos de todas las cuentas, sino que son considerados "estándares" del proyecto o cuenta de la que se trate.

8. Estrategias. Dentro de las estrategias implementadas se encuentran:

- Determinación de estándares de interface y estándares de codificación. Estipulados por cuenta o por proyecto. Los estándares de interface se refieren a la presentación que percibe el usuario de las distintas pantallas que conforman los sistemas. Por ejemplo, la posición de los botones de mantenimiento de datos en la pantalla, el tamaño de los botones, de las letras y de todos los elementos que conforman una pantalla, los colores, el despliegue de mensajes, etc. La utilidad de los estándares de codificación es para los desarrolladores, pues tienen como objetivo facilitar el mantenimiento del código y en ocasiones estipular el comportamiento de los programas. Entre ellos pueden encontrarse la nomenclatura para nombrar archivos y programas, la indentación del código, el control de bloqueos de registros, el manejo de mayúsculas y minúsculas en el código, la separación del funcionamiento en distintos procesos, la nomenclatura de nombres de variables y parámetros, etc.
- Aplicación de conteo de puntos de función. El conteo de puntos de función es una técnica para la medición de software cuyas bases radican en medir el tamaño de las aplicaciones de acuerdo a las operaciones reconocidas por el usuario que contiene el sistema. Esta técnica es utilizada para llevar a cabo la estimación del presupuesto y del costo de los sistemas. Los ingenieros capacitados para realizar los conteos son relativamente pocos, por lo que cuando se llegan a requerir los totales de puntos de función de varios sistemas o proyectos existen problemas de asignación, pues el conteo no es su única actividad y esto ocasiona que se carguen a una o más personas más responsabilidades de las que pueden cubrir en su horario normal.
- Reporte diario de actividades. Explicado a detalle en el capítulo anterior.

Cabe mencionar los distintos tipos de estrategias que debe definir toda empresa y que certum se ha ocupado de implementar, pero no se detallarán por ser parte de otro tipo de análisis. Dichas estrategias son: comercial, de producción, de compras, de personal y financiera⁷.

⁷ SALDIVAR, Antonio. "Planeación financiera de la empresa" Ed. Trillas pag. 21-25

2.1.2 Aplicación de planeación estratégica

La planeación estratégica atrajo el interés de las organizaciones durante los setenta, época en la que se produjeron en Estados Unidos problemas de recesión, inflación sin precedentes, escasez de petróleo y una incrementada competencia mundial para muchos productos. Esta situación dificultó demasiado la planeación exacta al mismo tiempo que se incrementó su importancia; es así como se hizo necesario el desarrollo de una serie de herramientas para la planeación estratégica. La administración evidenció el hecho de que las organizaciones que usaban bien la planeación estratégica superaban a aquellas que no lo hacían.

A continuación se desarrolla un análisis para poder determinar si certum aplica la planeación estratégica, revisando el seguimiento de los nueve pasos que la conforman.

PROCESO DE PLANEACIÓN ESTRATÉGICA

El proceso de planeación estratégica está definido pensando en la planeación de toda una organización. El siguiente, es un análisis del uso de planeación estratégica durante la planeación y desarrollo de los proyectos en certum, mas no de la planeación general de la organización para realizar ventas y desarrollarse como empresa. Por esta razón, en los pasos del proceso se sustituye la palabra "organización" por "proyecto".

1. Definir la misión del proyecto.

La misión es definida inicialmente por el cliente al establecer lo que requiere que el sistema haga. Por lo general no se cuenta con requerimientos por escrito y muy probablemente la explicación verbal suele ser confusa y no lo suficientemente clara como para comenzar a diseñar el sistema. Es labor de certum aclarar todas las dudas y proponer soluciones a los problemas del cliente para poder contar entonces con una misión clara del sistema.

2. Establecer objetivos.

Los objetivos de todos los proyectos son compartidos con los objetivos generales de la empresa (determinados como *externos* anteriormente):

- Asegurar la máxima prosperidad tanto para los inversionistas como para las personas que laboran en la empresa.

- La permanencia de la empresa en el mercado mediante el cumplimiento de los requerimientos del producto (calidad), del precio y del servicio, siguiendo ese estricto orden.

3. Analizar los recursos del proyecto.

No solo se trata de enumerar los recursos humanos, financieros y físicos, además de las capacidades que la organización posee, sino de vislumbrar su *ventaja comparativa*. La ventaja comparativa se refiere a la competitividad relativa que la organización tiene sobre sus competidores presentes y futuros.

La ventaja de certum radica en LA TRANSPARENCIA que ofrece a sus clientes. Mostrar a los clientes los procesos, el avance de los proyectos, los problemas que se presentan, las soluciones utilizadas para atacar dichos problemas, y en general, el estado en el que se encuentran los proyectos, da una excelente imagen de seguridad y automáticamente genera confianza entre el proveedor y el cliente.

Dicha transparencia se hace presente a través de:

- El reporte diario de actividades. Los reportes de SILPRO que se mencionaron en el primer apartado de este trabajo, así como toda la información necesaria que se deriva del mismo, es presentada al cliente como los datos reales del esfuerzo que ha significado el desarrollo del proyecto y por lo tanto de la factura que avala la cobranza. El método de cobro se realiza de acuerdo a lo acordado en el contrato, cuyos puntos pueden variar de una cuenta a otra, pero todos los contratos tienen en común el hecho de tomar como referencia las horas reportadas. Por esta razón, la información capturada por cada uno de los ingenieros debe ser totalmente confiable; el aseguramiento de dicha confiabilidad es realizando el reporte DIARIO, pues nadie puede tomarse el lujo de dejar pendiente el reporte de una semana o incluso de un par de días, aún si se presume de contar con una memoria privilegiada. Es conveniencia también de los ingenieros el indicar su trabajo real, pues al realizar el corte mensual de SILPRO (actividad realizada por el departamento de Contabilidad), cada ingeniero debe cubrir el mínimo de horas reportadas de acuerdo a los días hábiles del mes, tomando en cuenta que un día normal está compuesto por ocho horas de trabajo. Cualquier desviación de estas ocho horas debe ser reflejada en el reporte. Tomando en cuenta que uno de los valores con los que debe contar toda persona que forme parte de certum, es la HONESTIDAD, si un ingeniero, por cualquier motivo o razón, trabajara más o menos horas en un día, esta situación debe ser reportada. Pero no solo es gracias a la honestidad el que el registro sea real, sino a que los ingenieros están concientes de lo que implican las falsedades en estos datos, pues es de su conocimiento que:

Es gracias a esta información que su esfuerzo puede ser valorado si es que han dedicado a su trabajo más tiempo del reglamentado.

Es el origen del costo de los proyectos, se trata de información que más tarde será utilizada para llevar a cabo estimaciones de costos de actividades e incluso de proyectos completos, y

Se trata de la *ventaja comparativa* con la que cuenta la empresa de la que forman parte, y de eso depende el crecimiento de la empresa, y así, de sus propios intereses al ser integrantes de ella.

- El panel de control. El panel de control consta de la esquematización del plan del proyecto elaborado mediante la herramienta Microsoft Project. El esquema contiene cada una de las actividades contempladas para la realización del proyecto en su totalidad (o hasta el alcance que haya sido definido, por ejemplo, solo el diseño, o la construcción de uno de sus módulos), el tiempo estimado que requiere cada actividad para ser realizada y la persona asignada a la misma. Las actividades consideradas como ruta crítica son resaltadas con otro color. Es común que se les invite tanto a los clientes como a aquellos con los cuales la firma del contrato aún está pendiente, a que visiten las instalaciones de la empresa y es obligación de cada responsable de proyecto, que su respectivo panel de control se encuentre a la vista, colocado en una de las pizarras que están a disposición de cada proyecto o cuenta. Otro de los elementos del panel es la bandera de estado, que como su nombre lo indica, refleja el estado en el que se encuentra el Proyecto. La bandera debe estar colocada sobre la esquina superior derecha de la pizarra donde se encuentre ubicado el panel e indica el estado de acuerdo a su color.

Verde. Indica que todo marcha de acuerdo a lo planeado. El avance es normal y sin problemas que ameriten una mayor atención de los integrantes.

Amarillo. Mensaje de alerta. Los tiempos están rebasando los límites que se consideren normales y las fechas de inicio de actividades se han estado retrasando debido a la postergación de las fechas de fin de otras de las que dependen. Es necesaria la atención del responsable del proyecto y comenzar a investigar la razón de esta situación para poder atacarla.

Rojo. La fecha prometida de liberación del proyecto está en peligro de no poder ser cumplida. Es momento de aplicar medidas drásticas para ajustarse a los tiempos y fechas.

Negro. El aplazamiento de la fecha final es inminente. El proyecto no puede ser terminado o debidamente probado debido a la existencia de demasiados errores en la operación. Se debe recurrir a la negociación de una nueva fecha de entrega.

- Curva de incidentes. Se ha mencionado el registro y seguimiento que se le da a los errores de operación en los proyectos, y a los nuevos requerimientos solicitados por el cliente y en ocasiones propuestos por alguno de los miembros del desarrollo del proyecto. Durante el periodo de pruebas formales, diariamente es generada una gráfica que indica la existencia de incidentes reportados tomando en cuenta su severidad. Mientras la curva no sea estable, es decir, no guarde una forma de línea recta, indicador de la no existencia de incidentes que impidan la liberación del proyecto, la fase de pruebas no puede darse por terminada.

4. Examinar el ambiente.

Examinar el ambiente de un proyecto resulta mucho más sencillo que examinar el ambiente de la organización, pues en la segunda habrá que identificar los diversos factores políticos, sociales, económicos y de mercado; a diferencia del primero, en el que el ambiente lo constituyen los demás proyectos que se encuentren en marcha dentro de la propia organización. Así, tanto el encargado del proyecto como el encargado de producto, se hacen cargo de investigar si es necesario y/o posible obtener recursos de otros proyectos (intercambio de ingenieros o hacer uso de herramientas y técnicas anteriormente utilizadas dentro de la empresa, por ejemplo). El intercambio o reasignación de personas entre proyectos es muy común, pues al dar por terminado uno, sus participantes deben ser reasignados a otros proyectos, ya sean nuevos o existentes. Debido a que dentro de una misma cuenta pueden existir varios proyectos, en estos casos la rotación de ingenieros es más sencilla, pues los productos de una cuenta están relacionados y las herramientas de desarrollo son las mismas. Es evidente la necesidad de contar con una relación actualizada de ingeniero-proyecto, así como la duración de cada proyecto y las entregas pendientes para poder formar los equipos de trabajo en el tiempo adecuado y con el número y tipo de personas adecuados.

- 5. Hacer predicciones. Las predicciones sobre cómo se va a desenvolver el desarrollo de cualquiera de los servicios ofrecidos, es elemental para tener éxito en los mismos. En el caso de certum, la predicción más importante, y al mismo tiempo complicada, es la estimación de costos y definición de fechas de liberación. Este proceso se explicará con detalle en el apartado 2.1.3.
- 6. Oportunidades y riesgos. El hecho de contar con el panel de control, constituye una fuente de información con la cual los miembros del equipo de desarrollo analiza las oportunidades y los riesgos. Los ingenieros pueden enterarse de las fases de las que consistirá el desarrollo, el tipo de actividades incluidas en el plan y las herramientas de desarrollo contempladas. Esto les permite hacer del conocimiento del responsable del producto, su interés en alguna de las actividades contempladas. Para tener presentes todos los riesgos que cualquier integrante del equipo detecte, se cuenta con un sistema desarrollado por certum en Lotus Notes, denominado "Base de datos de Asuntos". Existe una

base de datos para cada proyecto, algunas generales como "Tecnología en certum" y otras específicas de cada cuenta, por ejemplo "Nombre de la cuenta Asuntos con cliente". En estos sistemas, se sigue el siguiente proceso:

- Se levanta un asunto. El título por lo general se trata de una pregunta sobre cómo atacar un problema en particular, por ejemplo: "¿Qué propuestas se deben incluir en la especificación del sistema de Seguridad?. Se incluyen las personas involucradas en el asunto. Al guardarlo, el tema se envía por e-mail a los involucrados.
- Inicia la participación de los involucrados anexando "ideas" que respondan a la pregunta o al tema del asunto. Cada respuesta se envía por e-mail anexando la liga correspondiente al asunto.
- Se agregan puntos a favor y puntos en contra a cada una de las ideas.
- Es posible también agregar otros textos a cada idea y postura, en forma de "notas".

Las BD's de Asuntos constituyen así, una fuente de problemas con sus posibles soluciones e ideas. Pueden ser consultadas fácilmente y permiten contar con una visión amplia de los riesgos presentes en cada proyecto, cuenta o incluso en la empresa en general.

7. Identificar y evaluar estrategias alternativas.

Una estrategia importante es el uso de herramientas adecuadas no solo para diseñar y construir, sino para llevar a cabo otras actividades necesarias para la organización y coordinación del proyecto. Por ejemplo, el manejo de un control de versiones de archivos. Hasta hace 6 meses, se utilizaba "PVCS", un software capaz de controlar versiones que operaba bajo ambiente MS-DOS. La herramienta era útil pero se comenzaron a detectar problemas con las computadoras que operan Windows 2000. No existe una nueva versión del software que arregle este problema, sino que quedó obsoleto en ese momento y el producto se declaró discontinuado. En ese momento comenzó un proceso de evaluación de otras herramientas que ofrecieran el mismo tipo de servicio, pero se buscó que trabajaran en ambiente gráfico (Windows). Se analizaron 3 de las que existen en el mercado tomando en cuenta la confiabilidad de las versiones, la capacidad para explotar la información como las diferencias entre cada una de las versiones, quién hizo modificaciones, cuándo e incluso a qué hora, así como la facilidad para operarlo y costo, entre otros puntos.

8. Seleccionar e instrumentar la estrategia

De acuerdo a los resultados de los análisis del punto no. 7, se selecciona la estrategia a seguir. Continuando con el ejemplo del sistema controlador de versiones, una vez seleccionado se procede a instalarlo en todas las computadoras de los ingenieros que construyen. En este caso, si bien se trata de un punto delicado en cuanto al proceso de desarrollo de sistemas, no es el de mayor prioridad, por

lo que no es posible asignarle demasiado tiempo a la capacitación de los usuarios. Se proporciona una plática sobre el funcionamiento general y los usuarios se encargarán de ir conociendo el resto durante la práctica. En este caso, la instrumentación se decidió así pero si se tratara por ejemplo de una nueva herramienta de desarrollo o manejador de base de datos, por supuesto que se asignaría mucho más tiempo a la capacitación y también la evaluación de las estrategias (herramientas) sería más detallado.

Además de los puntos anteriores, "la planeación estratégica debe delinear los valores de la organización, porque deben de influir en la forma en que ésta trabaja y logra sus objetivos"⁸. *certum* cuenta con su lista de valores con los cuales se ha definido que debe contar cada persona que labore dentro de la empresa. Dichos valores son:

- Servicio y Respeto al cliente
- Compromiso - Responsabilidad - Puntualidad
- Honestidad
- Transparencia
- Flexibilidad - Rigor
- Orgullo Ingenieril - Productos Sencillos
- Consistencia - Coherencia
- Pensadores Estratégicos
- Reto

2.1.3 Proceso de estimación

Si partimos de que "el desarrollo individual tiene lugar más eficazmente cuando los individuos participan en la toma de las decisiones que los afectan"⁹, es bien aceptado el proceso de estimación de horas que se requerirán para la realización de los proyectos, mismo que es presentado a continuación.

Se ha presentado como parte de la visión de SERINF, el cumplimiento de la liberación de los productos con un total de horas invertidas en el mismo lo más semejante posible a las horas estimadas al inicio del proyecto. Esta estimación no se realiza una sola vez sino que es reevaluada dos veces más. La primera estimación es llevada a cabo por el ejecutivo de la cuenta, quien dentro de la jerarquía, se encuentra inmediatamente después del director general (ver organigrama en el Anexo D). El ejecutivo de la cuenta hace uso de la historia de los proyectos registrada en SILPRO así como en los archivos generados en MS Project que contienen todas las actividades llevadas a cabo, quién las realizó y en cuánto tiempo.

La experiencia es un factor importante, pues gracias a ella, el ejecutivo puede contar con un panorama general del tipo de proyecto a realizar y a partir de él analizar las fases potenciales con las que contarán el diseño y la construcción, pero los factores que mayormente influyen son¹⁰:

⁸ <http://www.aceproject.org/main/espanol/em/emd01.htm>

⁹ ACKOFF, Russell L "Planificación de la empresa del futuro" Ed. Limusa México, pag.41"

- El grado de automatización
- La complejidad conocida del sistema
- El número de interfaces a desarrollar
- Los estándares de operación

Con este primer estimado, que se establece puede tener una variación de hasta un 50%, junto con la presentación del esquema de trabajo estándar de certum (comercialización) y el "análisis de operaciones" (A de O), se lleva a cabo la venta de servicios.

Grosso modo, el A de O consiste en las pláticas y reuniones necesarias entre SERINF y el cliente para detallar lo que se espera del sistema, a partir de los cuales será posible definir los valores de los factores mencionados.

Con el compromiso del primer estimado se comienza a establecer el grupo de personas que se asignarán al proyecto, el tipo de tecnología que se utilizará, si se requerirá de programas de capacitación, nuevas contrataciones, el alcance del proyecto y todo lo que sea relevante para comenzar la fase de diseño. En este proceso participa tanto la persona asignada para llevar a cabo el control del proyecto, como el "responsable del producto". Una vez decidida la tecnología que se utilizará, la cual abarca el equipo de cómputo con el cual se desarrollará, el lenguaje de programación, el manejador de base de datos, la infraestructura con la que cuenta el cliente y la que necesitará para hacer uso del producto que se entregue, se nombra a una persona que funcione como "experto técnico". El experto técnico debe tener un amplio conocimiento de la tecnología a utilizar y de los riesgos que implique el uso de la misma para poder apoyar a los desarrolladores con los problemas que se presenten, o en su defecto, ponerse en contacto con el departamento de soporte técnico de los proveedores de las herramientas de desarrollo.

En este momento se da inicio a la "construcción del demo del sistema", fase que pertenece al "servicio de diseño" (S de D) y que consiste de una presentación del menú principal del mismo así como de las distintas pantallas y procesos de los que esté formado. De dicho demo depende en gran medida la segunda estimación, pues se contará con una idea clara de todas las partes de las que estará formado el sistema, su prioridad para la operación general y el grado de dificultad de construcción de cada una de ellas. Con la visión más amplia que proporcionan los participantes de la elaboración del demo más la opinión de las personas que el encargado del producto considere necesarias, se da a conocer al cliente el segundo estimado de tiempo de terminación del proyecto.

¹⁰ certum, 1994 "Guías para la planeación y control de proyectos" pag. 11

En el diseño del sistema participa especialmente el encargado del producto, respaldado por el equipo que haya sido elegido para la realización de las actividades programadas.

Una de ellas, es el diseño del modelo físico de datos sobre el cual estarán basadas todas las transacciones de datos (creación, eliminación y modificación o comúnmente conocidas como operaciones ABC – Altas, Bajas y Cambios de datos) y procesamiento de la información.

Como se mencionó en el primer capítulo donde se desglosaron los módulos de los que está formado el SILPRO, los sistemas se dividen en funciones. La persona que realiza el diseño de una función no siempre se trata de la misma que la construirá. La primera intención es que sea así pero en ocasiones no es posible. Esto implica que la persona asignada para el desarrollo necesite de una inducción a la función a construir, ya sea por parte del encargado del producto, o bien, de quien realizó el diseño. Dicha inducción, el tiempo necesario para construir cada una de las funciones, las actividades de integración de las mismas en el sistema y procesos de pruebas, forman parte del tercer y último estimado. A partir del gran total de horas presupuestadas, se fija una fecha de entrega del producto.

El proceso de estimación presentado, se refiere a toda la realización de un proyecto de sistemas de cómputo, pero cabe mencionar que cada una de las partes que lo conforman (demo, diseño, construcción, integración, pruebas formales), es también presupuestada y vendida al cliente. Se cuenta entonces con un plan específico para cada una de ellas registrado en SILPRO y reflejado en la herramienta Microsoft Project para su control.

De esta manera, se observa que todas y cada una de las personas que formarán parte del proyecto están involucradas, en mayor o menor medida pero todas de manera importante, en el proceso de planeación del proyecto, del cual dependerá cumplir con la misión de la empresa.

Explícitamente se indicó la participación de los diferentes niveles de puestos en la estimación del presupuesto y promesa de fecha de entrega; la calidad del producto resulta del adecuado control del plan, así como de muchos otros factores que se analizarán a lo largo de este trabajo. Este tipo de mecanismo obedece a la teoría de que "el problema de la planeación no consiste en averiguar cómo se puede mejorar la vida de los demás, sino en cómo capacitarlos para que ellos mismos hagan esto por su cuenta, en forma más eficiente"¹¹, pues es responsabilidad de cada persona que el presupuesto que presente para terminar cualquier tipo de actividad asignada, se respete. De esta forma, todo individuo dentro de la empresa está obligado a aprender a estimar el costo de su trabajo.

¹¹ Idem

Desgraciadamente para la mayoría del personal, su única fuente de aprendizaje es la experiencia propia, pues no es extraño que suceda que la tarea de primera impresión parece muy sencilla pero al comenzar a realizarla se presenten cualquier tipo de problemas no previstos que no permiten terminar en el tiempo establecido. Sin embargo, no hay que perder de vista la pirámide de prioridades:



Como se podrá observar, los tres son sumamente importantes porque significan un fuerte compromiso con el cliente, pero siempre valdrá más un producto con calidad entregado después de la fecha prometida que uno que no funcione entregado a tiempo. Existen casos en los que ambos resultan tener la misma importancia por lo que el que tendrá que sufrir las consecuencias de retrasos y/o problemas funcionales del producto, será el Costo. Un ejemplo es el caso de un contrato de servicios que consistió de un sistema encargado del escrutinio de votos para elecciones de cargos políticos. El sistema debía estar a tiempo para el día de las elecciones y por ningún motivo se aceptaría el más mínimo retraso; en primer lugar por el tipo de evento del que se trataba y en segundo porque, para seguridad del cliente, había sido necesario firmar un contrato con certum que permitía cierta penalización legal en caso de no cumplir con la fecha estipulada. En esta ocasión (que no ha sido la única), fue necesario pedir a los ingenieros que dedicaran bastantes horas extras de trabajo, entendiéndose, salir de la oficina a altas horas de la noche y presentarse a trabajar en días festivos y fines de semana, pues cualquier retraso en la entrega o cualquier tipo de falla en la funcionalidad podría reflejarse en serios problemas para la empresa.

Este proceso es el que comúnmente se sigue para la estimación de costos y se ilustra con la figura 2.1.1, pero el hecho de que se definan el 1o, 2o. y 3er. estimado no quiere decir que forzosamente exista una sola secuencia 1-2-3 o que deban hacerse los tres; esto lo ordenará la certidumbre de las estimaciones anteriores y el comportamiento del desarrollo.

FALTA

PAGINA

57

SERVICIO	ESTIMADOS	UNIDAD	VARIACIÓN	OBSERVACIONES	PLAN DE ACTIVIDADES
Comercialización, previo al contrato.	Estimado preliminar de servicios ofrecidos	Mes	N/A	Estimación con base en experiencia o asignación presupuestal.	Se planea a nivel servicios completos
Análisis de operaciones [A de O]	Presupuesto/cota para A de O.	Horas	+25%	Negociado con cliente/usuario.	El subproyecto A de O al máximo detalle en el plan
	Primer estimado del proyecto	Mes	+50%	Estimación con base en procesos y grado de automatización (número de operaciones computarizadas), complejidad conocida (con base en procedimientos), interfaces, estándares de operación, etc.	Un subproyecto por servicio solo A de O con reales.
Diseño de sistemas [D de S]	Presupuesto/cota para diseño de sistemas	Horas	+25%	Negociado con cliente/usuario.	El subproyecto diseño de sistemas al máximo detalle en el plan
	Segundo estimado del proyecto	Quincena	+25%	Estimación con base en el diseño funcional y prototipos	A de O y D de S con reales
Construcción de sistemas [C de S] y pruebas formales (PF)	Presupuesto para ingeniería de detalle (IngDet)	Horas	+15%	Estimación con base en el diseño funcional. Estimación y control interno.	El subproyecto C de S se divide en dos: ingeniería de detalle y codificación. IngDet al máximo detalle en el plan
	Tercer estimado del proyecto	Día	0 %	Esta es la fecha compromiso para el bono de tiempo de entrega y la definitiva para el cliente.	Plan de trabajo al máximo detalle incluyendo pruebas formales

Fig. 2.1.1 Cuadro de estimación de costos de servicios

TESIS CON
 FALLA DE ORIGEN

2.2 Análisis del diseño y construcción de sistemas

La existencia de un proceso definido para llevar a cabo el desarrollo de aplicaciones, es esencial para la presentación de trabajos cuya organización, planeación, integración y exhaustivas pruebas, salten a la vista del cliente obteniendo así su confianza tanto en el producto como en el proveedor de software. Es así como surgen los diagramas de diseño y construcción de sistemas que fueron presentados en el primer apartado del presente trabajo (pags. 12-13).

2.2.1 El diseño de sistemas

Las secciones de Inicio, Contexto y Cierre de Servicio indicadas en el diagrama correspondiente no forman parte de este análisis, pues la atención se centrará en el Diseño Funcional. La razón es, que esta etapa conforma la parte medular del diseño, pues se ocupa de definir:

- El proceso de recopilación de información sobre los requerimientos del sistema. "Un requerimiento es una característica que debe incluirse en un nuevo sistema y puede consistir en una forma de captar o procesar datos, producir información, controlar una actividad de negocio o dar apoyo a la gerencia"¹²

Se debe tener cuidado de contar con todos los requerimientos antes de comenzar cualquier diseño. Se puede tomar como referencia la tipología de James Senn:

- Básicos
 - De transacciones de los usuarios
 - De decisión de los usuarios
 - Para toda la empresa
- La interfaz presentada al usuario, a través de la cual se capturarán los datos de entrada
 - Lo más importante, la estructura bajo la cual se almacenará la información en la base de datos, de manera que puedan ser explotados y manipulados optimizando recursos y ofreciendo el mejor *performance* posible.
 - En caso de que se requiera, definir la planeación de prototipos de algunas de las funciones, para comprobar que lo diseñado sea posible de ser construido mediante las herramientas elegidas (Front-End). "Un prototipo es un sistema de trabajo que se desarrolla con rapidez para probar las ideas y el entendimiento sobre el nuevo sistema; en otras palabras, no sólo es un diseño en el papel, sino un software que corre y produce información impresa o en pantalla."¹³

¹² SENN James A. "Análisis y diseño de sistemas de información" Ed. Mc Graw Hill, México 1990 p.70

¹³ Idem p.238

Se puede determinar entonces, que los pasos más importantes en el Diseño Funcional son:

- Diseñar y estimar funciones
- Modelar datos (lógico)
- Modelar datos (físico)
- Integrar datos, funciones y requerimientos

El tipo de análisis para realizar estas actividades difiere de acuerdo a las técnicas y herramientas que se utilicen, y actualmente en certum, no existen estándares sobre cuáles usar. Esto ocasiona que el diseño y documentación de los proyectos sean distintos entre ellos en cuanto a forma.

En el siguiente capítulo, se expondrá una técnica de modelado de datos que ayudará a crear un sencillo mapeo al modelo físico (estructura de la base de datos) que incluya todos los aspectos de reglas de negocio y procesos de operación del cliente.

El estandarizar el uso de una técnica de diseño de sistemas en todos, o la mayoría de los proyectos de la empresa, creará poco a poco el dominio de la misma y constituirá una herramienta clave para minimizar riesgos.

El impacto mayor de su utilización será el COSTO, pues el no contar con un diseño robusto desde el inicio ocasiona graves consecuencias que pueden ser detectadas durante la construcción, o peor aún, hasta la fase de pruebas, provocando cambios estructurales en la base de datos que pueden resultar fatales, pues puede afectar el código de varias pantallas. Cuando esto llega a suceder, se debe retroceder en el proceso y volver a pasar por fases de documentación, codificación y pruebas.

2.2.2 Construcción de sistemas

Parte importante de la construcción radica en la documentación que se mantiene sobre la organización del código y el uso de objetos de la base de datos, de manera que pueda constituir una herramienta que facilite el mantenimiento de los sistemas. Algunos de los aspectos que debe cubrir son:

- Qué objetos de la base de datos son consultados, utilizados y/o modificados por cada una de las funciones del sistema. Como se ha explicado con anterioridad, una función del sistema se conoce, por lo general, como una pantalla que realiza una función específica.
- Qué operaciones son ejecutadas por cada función. De esta manera, es posible identificar aquellas que sean compartidas entre funciones, y si esta situación se detecta en el momento adecuado, será posible el reuso de código. El reuso es conveniente porque si su buen funcionamiento es probado una vez, puede ser utilizado sin riesgo, además de que el mantenimiento se simplifica y el espacio en memoria es bien aprovechado.

La dificultad de estos dos puntos radica en el momento en que son detectados. Actualmente, parte del proceso de desarrollo es crear esta documentación, pero se realiza una vez que el producto está casi terminado, sirviendo de apoyo para la fase de mantenimiento mas no para la de construcción.

El contar con este tipo de información desde el inicio de la construcción, sería de mucha utilidad para los desarrolladores, pues evitaría la duplicidad de código y economizaría tiempo, además de que la integración sería mucho más transparente. La integración se refiere a la fase en la que se reúnen todas las funciones y módulos, conformando así el producto final.

Se dice que la integración puede realizarse de manera *transparente* cuando no requiere de mucho más trabajo adicional que la organización de los distintos módulos en el menú principal del sistema, y la creación del paquete de instalación.

Se ha detectado en certum, que por lo general, la fase de integración se alarga mucho más tiempo del estimado, y esto es en gran parte provocado por una falta de organización en el proceso de construcción, pues es acostumbrado que las funciones son asignadas a los ingenieros y cada uno de ellos se enfoca simplemente a la construcción de su función, sin tomar en cuenta su integración con el resto del módulo, y mucho menos con el sistema en general. Para corregir este comportamiento, no solamente hace falta inculcar una actitud diferente en el personal que codifica, sino también diseñar y promover el uso de herramientas que ayuden a tener documentado el avance de la construcción, mismas que se expondrán en el siguiente capítulo.

Pero no sólo gracias a la documentación se logra una construcción robusta con el mínimo de errores; lo principal es el nivel de conocimiento que tenga el programador sobre la herramienta con la que se construye. Para mejorar día con día este nivel, en ocasiones el trabajo diario no es suficiente, se necesita también de la difusión de "tips" encontrados durante la construcción por cada uno de los ingenieros y la ayuda que puedan ofrecer los más experimentados revisando directamente el código de los demás.

Estos sanos procedimientos son, en ocasiones, puestos en práctica por algunos proyectos, pero para garantizar que se tomen en cuenta como pasos esenciales del proceso de construcción, es necesario incluirlos en el diagrama oficial de dicho proceso. Más adelante, en la sección de mejoras al servicio de construcción de sistemas, se verán reflejadas estas propuestas.

TESIS CON
FALLA DE ORIGEN

Conclusiones

Para concluir este análisis, se hará uso de un cuadro que permita visualizar las Fortalezas, Oportunidades, Debilidades y Amenazas (FODA) latentes en el proceso de software ejercido por certum y que ha sido presentado a lo largo de esta investigación.

En cada uno de los puntos desglosados en el siguiente cuadro, se agrega un elemento para indicar el área de la política de calidad sobre la que tienen mayor impacto.

Simbología: (Q) Calidad
 (F) Fecha de entrega
 (C) Costo

FORTALEZAS	OPORTUNIDADES
<input type="checkbox"/> Mejora continua del diseño del plan de proyectos de acuerdo a la experiencia (QF)	<input type="checkbox"/> La empresa es reconocida como precursora del manejo del reporte diario de actividades (QFC)
<input type="checkbox"/> Conocimiento y aplicación de la métrica de Puntos por Función para medir el tamaño de los sistemas y el avance de la construcción de los mismos (C)	<input type="checkbox"/> Acceso a información acerca de la implementación de procesos de aseguramiento de calidad de software (SQA Software Quality Assurance) (ver sección 3.3.2) (Q)
<input type="checkbox"/> Base de datos de Asuntos. Contribuye al análisis y control de riesgos (Q)	<input type="checkbox"/> Mantenerse actualizados en cuanto a información sobre la situación del mercado internacional (QFC)
<input type="checkbox"/> Documentación disponible sobre los procesos en BD Servicios certum de acuerdo a las normas ISO-9001 (Q)	<input type="checkbox"/> Continuar participando en eventos de carácter informático, como el Primer Seminario Internacional de Calidad en Desarrollo de Software (ver sección 3.3) (Q)
DEBILIDADES	AMENAZAS
<input type="checkbox"/> Falta de una técnica de diseño de sistemas aplicable a todos los proyectos (FC)	<input type="checkbox"/> Falta de certificación por parte de los ingenieros en las herramientas de desarrollo, a diferencia de las empresas competidoras (Q)
<input type="checkbox"/> Tardía documentación de la organización del código y falta de revisión y retroalimentación sobre el mismo (FC)	<input type="checkbox"/> Considerable adelanto en cuanto a procesos específicos de calidad de software por parte de empresas extranjeras (hindúes, sobre todo) (Q)
<input type="checkbox"/> Falta de control de calidad en el seguimiento de procesos (Q)	<input type="checkbox"/> Es posible conseguir costos más bajos por los mismos servicios, en el extranjero (C)
<input type="checkbox"/> Defectuoso control de calidad en los productos (Q)	<input type="checkbox"/> Dado el decremento del nivel de productividad, éste ha dejado de ser un factor para diferenciar a la empresa del resto.
<input type="checkbox"/> Mala estimación (FC)	

Podemos observar que:

1. Las fortalezas radican en la aplicación de técnicas de procesos de software, prácticas de planeación y control, así como de certificación de calidad.

2. Se cuenta con las oportunidades planteadas gracias a las fortalezas, pues constituyen la base y un avance para continuar implementando técnicas reconocidas a nivel mundial para mejorar el proceso de desarrollo de software. Se hablará de la importancia y del provecho que pueden brindar dichas técnicas en la última parte de esta investigación.
3. Las debilidades encontradas se deben, no a la falta de diseño de procesos, sino a su pobre seguimiento, situación que puede ser superada aplicando los puntos indicados como oportunidades.
4. Las amenazas atacan directamente la política de calidad, pero para evitar su aparición y reducir la incertidumbre que provocan en los resultados, es que se ha trabajado en este análisis y que concluye a continuación con la exposición de las mejoras a los procesos de diseño, construcción, planeación y control de proyectos.

FALTA

PAGINA

65 |

CAPÍTULO III

Propuestas de estrategias para reducir la incertidumbre

Objetivo: *Con base en los resultados obtenidos a partir del análisis presentado en el segundo apartado del trabajo, se expondrá otro tipo de implementación de los servicios ofrecidos por parte de la empresa en cuestión. Se expondrá la probable permanencia en el tiempo de esta nueva implementación, tomando en cuenta el acelerado desarrollo de las herramientas y técnicas para el desarrollo de software.*

3.1 Servicio de diseño de sistemas

La calidad del diseño de una base de datos es crítica para el desarrollo de cualquier sistema de información. Tal y como una casa construida con base en un buen plan arquitectónico es más segura y funcionalmente habitable, una base de datos bien diseñada garantiza que la información será fácilmente obtenida y confiable.

A cualquier persona que se encuentre involucrada con el negocio de sistemas, le resultará familiar que después de que el cliente ha aceptado las propuestas de diseño e incluso algún tipo de demo, una vez que el sistema se ha construido hace observaciones como: *"Lo que quise decir fue y no que"* ó *"Yo le entendí que el sistema funcionaría de tal manera, y resulta que funciona de esta otra"* o peor aún... *"pensándolo bien, sería mejor si..."* cuando ese cambio de opinión implica cambios severos en la codificación en el mejor de los casos, pues en el peor de ellos, es necesario un cambio estructural en el modelo de datos. Un adecuado diseño que contemple la presentación y completa aceptación del cliente puede minimizar estos problemas.

Atacando directamente el primer defecto señalado en el análisis FODA presentado anteriormente, a continuación se propone el uso del Lenguaje de Modelado Unificado (UML - Unified Modeling Language), técnica que puede mejorar de manera considerable en certum el proceso de modelado de bases de datos y reducir la incertidumbre en los tiempos de entrega y costos.

3.1.1 El Lenguaje de Modelado Unificado (UML)

INTRODUCCIÓN AL UML

La programación estructurada nace para que los programas fueran más fáciles de manejar, ya que la complejidad con la que se construían provocaba grandes problemas al tratar de hacer modificaciones, ya sea para corregir errores, o para implementar nuevos requerimientos. La facilidad de esta metodología radicó en la separación de procesos en procedimientos y funciones especializadas de manera que pudieran ser reusables. A raíz de esto, se llevó la programación estructurada al análisis y al diseño.

A través del tiempo se fue mejorando el hardware y el software, lo que dio origen a utilizar con mayor facilidad los gráficos y se observó que la técnica de programación estructurada se podía aplicar con la visión de objetos.

El concepto de "objeto" en la jerga del desarrollo de software, nace cuando se comenzó a hablar de gráficos en los sistemas computacionales y se contó con una visión más clara del programa. Se ha demostrado que las técnicas orientadas a objetos son mejores que el resto de los enfoques para el manejo del ritmo de cambio cada vez más grande en las organizaciones actuales.

Al diseñar una base de datos para una aplicación en particular, se crea un modelo de la realidad concerniente al área de la aplicación. Técnicamente, se le conoce a dicha área de la realidad como "el universo de discusión" (UoD - *Universe of Discourse*)¹. La construcción de un buen modelo requiere de un conocimiento a fondo del UoD, y es una tarea realizada directamente por personas y no por máquinas. El principal reto es describir el UoD de manera clara y precisa; se requiere de sumo cuidado dado que los errores sembrados se filtran por las siguientes etapas del desarrollo de sistemas, y mientras más tarde se detecten, la corrección será más cara y complicada.

La persona responsable de modelar el UoD, es conocida como "modelador". Si el modelador está familiarizado con el área de la aplicación, puede trabajar solo, pero por lo general requiere de alguien que colabore con él y que conozca a fondo tanto el área como los requerimientos del sistema. Dicho de otra forma, el cliente debe participar muy de cerca en la primer etapa del diseño.

La principal característica con la que debe contar todo modelo, es poder lograr una descripción fiel del UoD haciendo uso del lenguaje natural, diagramas intuitivos y ejemplos, de manera que cualquier persona que conozca el UoD pueda interpretar el modelo y así trabajar junto con el modelador hasta llevar el modelo a un nivel conceptual. Una vez que el modelo sea comprensible y aceptado, podrá ser mapeado a cualquier *manejador de bases de datos* (DBMS - *DataBase Management System*) que haya sido elegido.

El modelado orientado a objetos abarca tanto el tipo de información del que se trate, como la conducta de objetos. El uso del Lenguaje de Modelado Unificado (UML - *Unified Modeling Language*) está principalmente enfocado al diseño de código para programas orientados a objetos (OO) pero ha recibido

¹ RUMBAUGH, JACOBSON, BOOCH "The Unified Modeling Language Reference Manual", Ed. Addison Wesley, EUA 1998, pag. 68

una rápida aceptación en el área de diseño de bases de datos; tanto, que ha sido adoptado por el Grupo de Manejo de Objetos (OMG – *Object Management Group*).

LOS INICIOS DE UML

A partir del año 1994, Grady Booch (precursor de la notación Booch) y Jim Rumbaugh (creador de *Object Modeling Technique*, OMT) se unen en una empresa común, "Rational Software Corporation", y comienzan a unificar sus dos métodos. Un año más tarde, en octubre de 1995, aparece UML 0.8. A finales de ese mismo año, Ivar Jacobson, creador de OOSE (Object Oriented Software Engineer) se añade al grupo. El modelo de Jacobson no representa muy bien las relaciones que existen en las bases de datos; su fuerte está en la técnica para describir los requerimientos de los sistemas de información.

Como objetivos principales de la consecución de un nuevo método que aunara los mejores aspectos de sus predecesores, sus protagonistas se propusieron lo siguiente²:

- El método debía ser capaz de modelar no sólo sistemas de software sino otro tipo de sistemas reales de la empresa, siempre utilizando los conceptos de la orientación a objetos (OO).
- Crear un lenguaje para modelado utilizable a la vez por máquinas y por personas.
- Establecer un acoplamiento explícito de los conceptos y herramientas.
- Manejar los problemas típicos de los sistemas complejos de misión crítica.

Lo que se intenta es lograr que los lenguajes de modelado que se aplican siguiendo los métodos más utilizados, sigan evolucionando en conjunto y no por separado, además de unificar las perspectivas entre diferentes tipos de sistemas de software, al aclarar las fases de desarrollo, los requerimientos de análisis, el diseño, la implementación y los conceptos internos de la OO.

MODELADO DE OBJETOS

Los modelos, al no ser una representación que incluya todos los detalles de los originales, permiten probar más fácilmente los sistemas que modelan y determinar los errores. Según se indica en la Metodología OMT (Rumbaugh), los modelos permiten una mejor comunicación con el cliente por distintas razones:

² Idem, pag. 75

- Es posible enseñar al cliente una posible aproximación de lo que será el producto final.
- Proporcionan una primera aproximación al problema que permite visualizar cómo quedará el resultado.
- Reducen la complejidad del original en subconjuntos que son fácilmente tratables por separado.

"La estandarización de un lenguaje de modelado es invaluable, ya que es la parte principal de comunicación. Si se quiere discutir un diseño con alguien más, ambos deben conocer el lenguaje de modelado y no así el proceso que se siguió para obtenerlo"³. Además, los modelos permiten corregir defectos antes de construir el producto y es posible descubrir detalles que se pasaron por alto en el planteo original de la idea.

Una de las metas principales de UML es avanzar en el estado de la industria proporcionando herramientas de interoperabilidad para el modelado visual de objetos. Sin embargo, para lograr un intercambio exitoso de modelos de información entre herramientas, se requirió definir a UML una semántica y una notación.

La notación es la parte gráfica que se ve en los modelos y representa la sintaxis del lenguaje de modelado. Por ejemplo, la notación del diagrama de clases define como se representan los elementos y conceptos como son: una clase, una asociación y una multiplicidad. ¿Y qué significa exactamente una asociación o multiplicidad en una clase?. Un metamodelo es la manera de definir esto (un diagrama, usualmente de clases, que define la notación). Para que un proveedor diga que cumple con UML debe cubrir con la semántica y con la notación.

Una herramienta de UML debe mantener la consistencia entre los diagramas en un mismo modelo. Bajo esta definición, una herramienta que solo dibuje, no puede cumplir con la notación de UML.

En OMT se modela un sistema desde tres puntos de vista diferentes donde cada uno representa una parte del sistema y una unión lo describe de forma completa. "En esta técnica de modelado se utilizó una aproximación al proceso de implementación de software habitual donde se utilizan estructuras de

³ BOOCH, JACOBSON, RUMBAUGH "Developing Applications with Visual Basic and UML", Ed. Addison Wesley, EUA 2000, pag. 37

datos (modelo de objetos), las operaciones que se realizan con ellos tienen una secuencia en el tiempo (modelo dinámico) y se realiza una transformación sobre sus valores (modelo funcional)⁴.

UML utiliza parte de este planteamiento obteniendo distintos puntos de vista de la realidad que modela mediante los distintos tipos de diagramas que posee. Con la creación del UML se persigue obtener un lenguaje que sea capaz de abstraer cualquier tipo de sistema, sea informático o no, mediante los diagramas, es decir, mediante representaciones gráficas que contienen toda la información relevante del sistema. Un diagrama es una representación gráfica de una colección de elementos del modelo, que habitualmente toma forma de grafo donde los arcos que conectan sus vértices son las relaciones entre los objetos y los vértices se corresponden con los elementos del modelo. Los distintos puntos de vista de un sistema real que se quieren representar para obtener el modelo se dibujan de forma que se resalten los detalles necesarios para entender el sistema.

⁴ <http://www.rational.com/uml/>

NOTACIÓN DE UML

Se necesita más de un punto de vista para llegar a representar un sistema. UML cuenta con diferentes tipos de diagramas gráficos para obtener estos distintos puntos de vista:

- Diagramas de Casos de Uso.

Los diagramas de casos de uso se utilizan para ilustrar los requerimientos del sistema al mostrar como reacciona una respuesta a eventos que se producen en el mismo. En este tipo de diagrama intervienen algunos conceptos nuevos: un *actor* es una entidad externa al sistema que se modela y que puede interactuar con él; un ejemplo de actor podría ser un usuario o cualquier otro sistema. Las relaciones entre casos de uso y actores pueden ser las siguientes:

- Un actor se comunica con un caso de uso.
- Un caso de uso extiende otro caso de uso
- Un caso de uso usa otro caso de uso

- Diagramas de Clases.

"Los diagramas de clases representan un conjunto de elementos del modelo que son estáticos, como las clases y los tipos, sus contenidos y las relaciones que se establecen entre ellos. Algunos de los elementos que se pueden clasificar como estáticos son los siguientes"⁵:

- Paquete: Es el mecanismo de que dispone UML para organizar sus elementos en grupos, se representa un grupo de elementos del modelo. Un sistema es un único paquete que contiene el resto del sistema, por lo tanto, un paquete debe poder anidarse, permitiéndose que un paquete contenga otro paquete.
- Clases: Una clase representa un conjunto de objetos que tienen una estructura, un comportamiento y unas relaciones con propiedades parecidas. Describe un conjunto de objetos que comparte los mismos atributos, operaciones, métodos, relaciones y significado. En UML una clase es una implementación de un tipo. Los componentes de una clase son:
 - o Atributo. Se corresponde con las propiedades de una clase o un tipo. Se identifica mediante un nombre. Existen atributos simples y complejos.
 - o Operación. También conocido como método, es un servicio proporcionado por la clase que puede ser solicitado por otras clases y que produce un comportamiento en ellas cuando se realiza.

Las clases pueden tener varios parámetros formales, son las clases denominadas plantillas. Sus atributos y operaciones vendrán definidas según sus parámetros formales. Las plantillas pueden

⁵ <http://gidis.ing.unlpam.edu.ar/personas/glafuente/uml>

tener especificados los valores reales para los parámetros formales, entonces reciben el nombre de *clase parametrizada instanciada*. Se puede usar en cualquier lugar en el que se podría aparecer su plantilla.

Relacionado con las clase se encuentra el término *utilidad*, que se corresponde con una agrupación de variables y procedimientos globales en forma de declaración de clase, también puede definirse como un estereotipo (o nueva clase generada a partir de otra ya existente) de un tipo que agrupa variables globales y procedimientos en una declaración de clase. Los atributos y operaciones que se agrupan en una utilidad se convierten en variables y operaciones globales. Una utilidad no es fundamental para el modelado, pero puede ser conveniente durante la programación.

Las clases se relacionan entre sí de distintas formas, que marcan los tipos de relaciones existentes: *Asociación, Composición, Generalización, Dependencia, Relación de refinamiento*.

- Diagramas de Comportamiento o Interacción.

Muestran las interacciones entre objetos ocurridas en un escenario (parte) del sistema. Hay varios tipos:

- Diagrama de secuencia. Muestran las interacciones entre un conjunto de objetos, ordenadas según el tiempo en que tienen lugar. En los diagramas de este tipo intervienen objetos, que tienen un significado parecido al de los objetos representados en los diagramas de colaboración, es decir son instancias concretas de una clase que participa en la interacción. El objeto puede existir sólo durante la ejecución de la interacción, se puede crear o puede ser destruido durante la ejecución de la interacción. Un diagrama de secuencia representa una forma de indicar el período durante el que un objeto está desarrollando una acción directamente o a través de un procedimiento.
- Diagrama de colaboración. Muestra la interacción entre varios objetos y los enlaces que existen entre ellos. Representa las interacciones entre objetos organizadas alrededor de los objetos y sus vinculaciones. A diferencia de un diagrama de secuencias, un diagrama de colaboraciones muestra las relaciones entre los objetos, no la secuencia en el tiempo en que se producen los mensajes. Los diagramas de secuencias y los diagramas de colaboraciones expresan información similar, pero en una forma diferente.
- Diagramas de actividad. Son similares a los diagramas de flujo de otras metodologías OO. En realidad se corresponden con un caso especial de los diagramas de estado donde los estados son *estados de acción* (estados con una acción interna y una o más transiciones que suceden al finalizar esta acción, o lo que es lo mismo, un paso en la ejecución de lo que será un

procedimiento) y las transiciones vienen provocadas por la finalización de las acciones que tienen lugar en los estados de origen. Siempre van unidos a una clase o a la implementación de un caso de uso o de un método (que tiene el mismo significado que en cualquier otra metodología OO). Los diagramas de actividad se utilizan para mostrar el flujo de operaciones que se desencadenan en un procedimiento interno del sistema.

- Diagramas de estado. Representan la secuencia de estados por los que un objeto o una interacción entre objetos pasa durante su tiempo de vida en respuesta a estímulos (eventos) recibidos. Representa lo que podemos denominar en conjunto una máquina de estados. Un *estado* en UML es cuando un objeto o una interacción satisface una condición, desarrolla alguna acción o se encuentra esperando un evento.

- Diagramas de Implementación.

“Se derivan de los diagramas de proceso y módulos de la metodología de Booch, aunque presentan algunas modificaciones. Los diagramas de implementación muestran los aspectos físicos del sistema. Incluyen la estructura del código fuente y la implementación, en tiempo de implementación. Existen dos tipos:”⁶

- Diagramas de componentes. Muestra la dependencia entre los distintos componentes de software, incluyendo componentes de código fuente, binario y ejecutable. Un componente es un fragmento de código software (un fuente, binario o ejecutable) que se utiliza para mostrar dependencias en tiempo de compilación.
- Diagrama de plataformas o despliegue. Muestra la configuración de los componentes hardware, los procesos, los elementos de procesamiento en tiempo de ejecución y los objetos que existen en tiempo de ejecución. En este tipo de diagramas intervienen nodos, asociaciones de comunicación, componentes dentro de los nodos y objetos que se encuentran a su vez dentro de los componentes. Un *nodo* es un objeto físico en tiempo de ejecución, es decir una máquina que se compone habitualmente de, por lo menos, memoria y capacidad de procesamiento, a su vez puede estar formado por otros componentes.

En el contexto de modelado de bases de datos, el diagrama a utilizar es el segundo. El diagrama de clases puede ser utilizado equivalente a un modelo entidad-relación, pero con lógica OO tomando en cuenta los siguientes puntos⁷:

⁶ http://gidis.ing.unlpam.edu.ar/personas/glafuente/uml/diagramas_de_implementacion.html

⁷ CARNEGIE MELLON UNIVERSITY. Software Engineering Institute. "The Capability Maturity Model. Guidelines for Improving the Software Process", Ed. Addison Wesley, USA, 1997, pag. 184

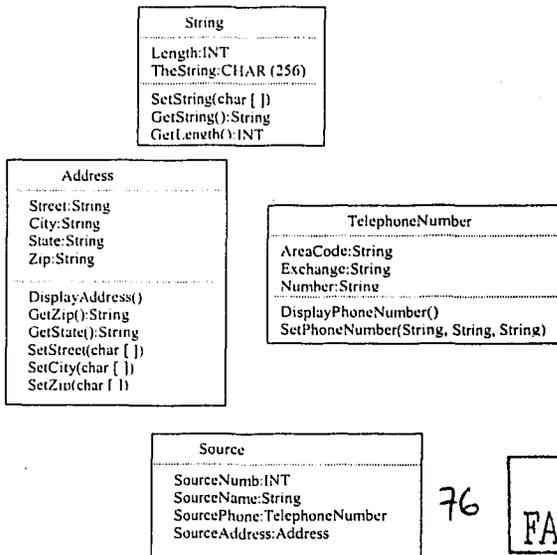
1. Una clase regular es representada por un rectángulo dividido en tres partes: nombre, atributos y métodos.
2. Una clase "madre" es representada por un rectángulo que contiene el nombre de la clase y dentro, los rectángulos de sus subclases. Por ejemplo, en la figura 3.1, las clases *Product* y *Source* están dentro de su clase "madre" correspondiente, *AllProducts* y *AllSources*.
3. Las asociaciones se muestran con líneas sin punta. La cardinalidad (multiplicidad) de una relación es expresada como: $n, n..n$ ó n^* . Por ejemplo, si la cardinalidad es 1, se escribe simplemente 1. Si la cardinalidad es 0 ó 1, se escribe 0..1. Si es cero o más, aparece como 0..*. Uno ó más se denotaría como 1..*. Nótese que en la figura 3.1 hay varias relaciones directas de tipo muchos-a-muchos, ilustradas con 0..* en alguno de los extremos de la línea de asociación.
4. La herencia es representada por una línea con punta de flecha sin relleno apuntando hacia la clase base. En la figura, las clases *Footwear* y *Headgear* cuentan con flechas de este tipo apuntando hacia la clase *Product*.
5. A lo que en las bases de datos relacionales se le conoce como "entidad compuesta", en UML se le llama "clase de asociación". Una clase de asociación está conectada a la relación a la cual aplica, mediante una línea punteada.

Además de las características básicas citadas, los diagramas UML pueden incluir cualquiera de las siguientes para sumarle funcionalidad:

1. Un atributo puede incluir información acerca de su visibilidad (pública, protegida o privada), tipo de dato, valor por default y dominio. En la figura 3.2 por ejemplo, se observan cuatro clases y el tipo de dato de sus atributos. Hay que tomar en cuenta que en el ambiente OO, los tipos de datos pueden ser otras clases. Así, la clase *Source* usa un objeto de la clase *TelephoneNumber* para su atributo *sourcePhone* y un objeto de la clase *Address* para su atributo *sourceAddress*. A su vez, *Source*, *Address* y *TelephoneNumber* contienen atributos que son objetos de la clase *String*.
2. Los métodos, oficialmente llamados "operaciones" en UML, pueden incluir su encabezado completo. En la figura 3.2 se observan los nombres de métodos seguidos de los tipos de datos que requieren

para ser ejecutados (parámetros). El nombre del método más sus parámetros, componen el encabezado del método. Si se devuelve un valor como resultado de la operación, se indica a continuación del encabezado con dos puntos (:) e inmediatamente después el tipo de dato del valor regresado, el cual puede ser un objeto de otra clase, o un tipo de dato común, como por ejemplo, un valor entero.

3. Las flechas con relleno pueden ser utilizadas al final de las asociaciones para indicar la dirección en la que la relación puede ser leída. Existen tres maneras posibles de utilizar flechas:
 - i. Usar puntas de flechas al final de todas las asociaciones hacia donde es posible la navegación. Si una asociación tiene un final sin punta, la navegación no es posible en esa dirección. Esto podría representar, por ejemplo, una relación no inversa entre dos objetos, donde solo uno de los dos objetos contiene el identificador del objeto relacionado.
 - ii. No utilizar ningún tipo de asociación, como en la figura 3.2. En este caso, el diagrama no provee información acerca de cómo puede ser navegado el diseño de la base de datos.
 - iii. No usar puntas en asociaciones navegables en ambas direcciones, pero usarlas cuando la navegación es posible en sólo una dirección. La desventaja de esta notación es que es imposible distinguir entre asociaciones navegables en ambas direcciones y aquellas que no permiten navegación. Ver figura 3.3.



76

TESIS CON FALLA DE ORIGEN

```
GetSourceInfo():String  
SetName(char [ ] )  
SetPhone(TelephoneNumber *)  
SetAddress(Address *)
```

Figura 3.2 Clases UML mostrando sus tipos de datos

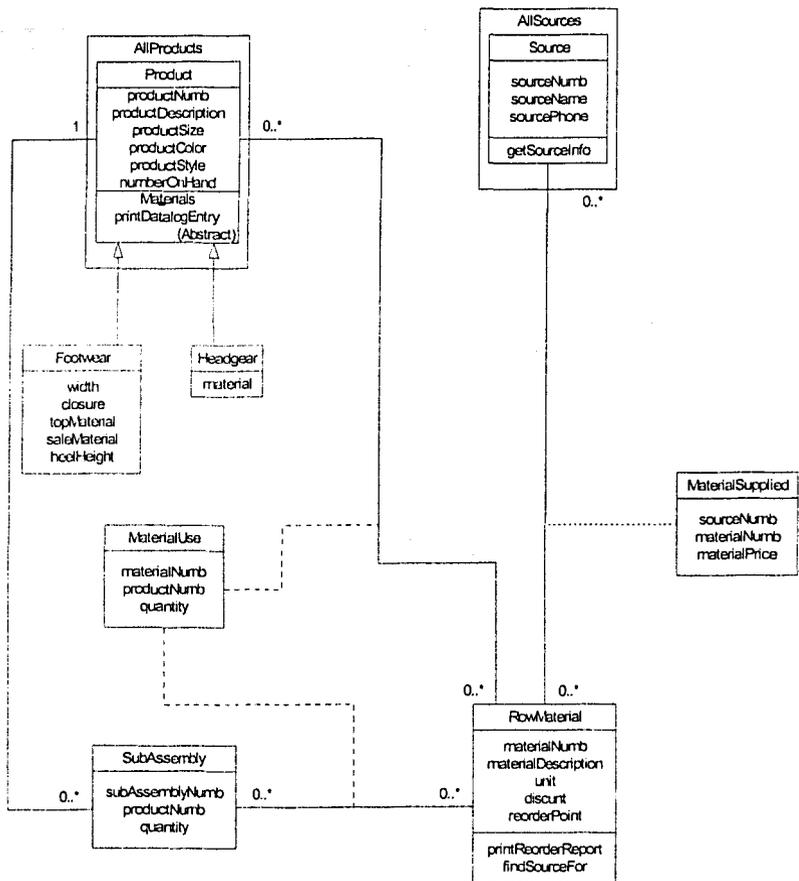


Figura 3.3 Ejemplo de un diseño de base de datos orientado a objetos en notación UML.

4. Una asociación que termina en forma de rombo o diamante relleno indica una relación tipo "contenido en". Por ejemplo, si se estuviera representando una hoja de cálculo, la relación entre ésta y sus celdas podría denotarse como se ilustra en la figura 3.4. El diamante relleno puede ser utilizado también para representar las clases agregadas, en lugar de hacerlo como se muestra en la figura 3.1.

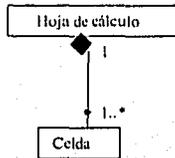


Figura 3.4 Usando UML para representar una relación de tipo "contenido en"

5. Cuando la asociación se da entre más de dos objetos, UML utiliza un rombo o diamante (sin relleno, o blanco) para representar la relación. Si una clase de asociación está presente, será conectada al diamante como se muestra en la figura 3.5. Las cuatro clases de la ilustración representan entidades en una base de datos de una sociedad de lectura de poesía. Una "lectura" ocurre cuando una persona lee una poesía escrita por uno o más poetas. La entidad *Reading* indica cuándo y dónde se llevó a cabo la lectura.

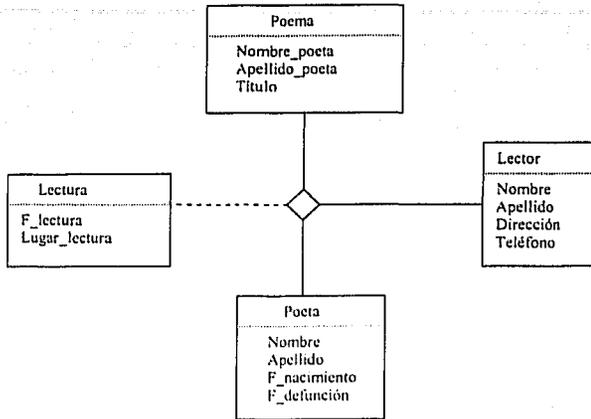


Figura 3.5 Usando UML para representar relaciones entre más de dos clases

Los cinco puntos anteriores representan únicamente una breve reseña de las bases para crear y/o comprender la notación utilizada en un modelo lógico de bases de datos, y junto con la pequeña presentación de sus inicios, objetivos y alcances, constituyen una invitación para profundizar en el tema, pues es tan amplio que requiere de la atención de toda una literatura, prácticas y ejemplos de implementación de acuerdo a la plataforma elegida.

A continuación se presenta una lista de los productos en el mercado para desarrollar diseños UML. Para mayor información sobre cada uno de ellos, consultar la página:

http://www.objectsbydesign.com/tools/umltools_byProduct.html

Compañía	Producto	Versión	Fecha	Plataforma	Precio
Computer Associates	AllFusion Component Modeler	4	Ene-00	Windows, Unix	n/a
XJ Technologies	AnyStates	3	Sep-02	Java VM	\$490
Tigris	ArgoUML	0.10.1	Jul-02	Java VM	\$0
Artiso	Artiso Visual Case	2.2	Sep-02	Java VM	\$449
BoldSoft	Bold for Delphi	3.2	Jun-02	Delphi	\$2900
Project Technology	BridgePoint	5	Abr-01	Windows, Unix	n/a
CanyonBlue	Cittera	1	Jun-01	Java VM	n/a
Codagen	Codagen Architect	2.5.2	May-02	Windows	n/a
Logic Explorers	Code Logic		Mar-02	Java VM	n/a
Select Business Solutions	Component Architect	4.1		Windows	n/a
Honeywell	DOME	5.3	Mar-00	Smalltalk	\$0
Atos Origin	Delphia Object Modeler (D.OM)	3.2.6	Dic-00	Windows	\$0
Embarcadero	Describe	5.8.2	Ago-02	Windows	n/a
Dia	Dia	0.88	May-01	Linux	\$0
Documentator	Documentator	4	Ene-02	Windows	\$89
Eldean AB	ESS-Model	2	Oct-01	Windows	\$0
Omondo	EclipseUML	1.1	Oct-02	Java VM	\$0
EctoSet	EctoSet Modeller	1.5.4	Abr-02	Windows	\$80
Elixir Technologies	Elixer CASE	1.2.4	Nov-99	Java VM	\$295
Sparx Systems	Enterprise Architect Professional	3.1	Ene-02	Windows	\$149
Tassc	Estimator / manager	4.1	Mar-01	Windows	\$4255
Novosoft	FL	0.5.5	Feb-02	Java VM	n/a
Fujaba	Fujaba	3	Jan-02	Java VM	\$0
Hoora	HAT Professional	3.1	Mar-01	Windows	\$800
Ideogramic	Ideogramic UML, Desktop Edition	2.3.2	Aug-02	Windows, Linux	\$1160

Ideographic	Ideographic UML, Whiteboard Edition	2.3.2	Aug-02	Windows	\$5195
Nasra Consulting	J2U	1.3.1	Jun-02	Java VM	\$499
Borland	JBuilder Enterprise	7	Jun-02	Java VM	\$2999
Oracle	JDeveloper	9i	Jan-02	Windows	\$0
Objective Ideas AB	JSequence	1	Aug-02	Java VM	\$199
Object Insight	JVISION	2.1	Oct-01	Java VM	\$499
Excel Software	Mac A&D Desktop	7.4	Jul-02	MacOS	\$1295
MagicDraw	MagicDraw UML Standard	5.5	Jul-02	Java VM	\$299
MagicDraw	MagicDraw UML Professional	5.5	Jul-02	Java VM	\$699
MasterCraft	MasterCraft Component Modeler	5	Jun-02	Windows	n/a
Mega International	Mega Suite	5.2	Oct-00	Windows	n/a
MetaMatrix	MetaBase Modeler	2.1	Mar-02	Windows	n/a
MetaCase Consulting	MetaEdit+	3		Windows, Unix	\$4500
Metamill Software	Metamill	2	Nov-01	Windows	\$85
ModelMaker Tools	ModelMaker	6.2	Aug-02	Delphi	\$269
Modelistic	Modelistic	1.1	Aug-00	Java VM	\$299
Novosoft	Novosoft UML Library	1.4	Dec-01	Java VM	\$0
Object Plant	Object Plant	3.0.1	Feb-02	MacOS	\$35
Object Domain Systems	Object Domain Standard	3	Nov-01	Java VM	\$495
OTW Software	Object Technology Workbench Private	2.4	Apr-00	Windows	\$795
Object Domain Systems	Object Domain Professional	3	Nov-01	Java VM	\$995
OTW Software	Object	2.4	Apr-00	Windows	\$1495

	Technology Workbench Team				
Telelogic	ObjectGeode	4.1	Jun-99	Windows, Unix	n/a
Softteam	Objecteering Personal Edition	5.1.0	May-01	Windows, Unix	\$0
Softteam	Objecteering Personal Edition / Java	5.1.0	May-01	Windows, Unix	\$859
Softteam	Objecteering Project Edition	5.1.0	May-01	Windows, Unix	\$1569
Softteam	Objecteering Enterprise Edition	5.1.0	May-01	Windows, Unix	\$2569
MicroTOOL	ObjectiF	4.6		Windows	n/a
TNI-Valiosys	OpenTool	3.2	Jan-01	Windows, Unix	n/a
Plastic Software	Plastic	3	Jan-01	Java VM	\$297
Gentleware	Poseidon for UML - Community	1.4	Sep-02	Java VM	\$0
Gentleware	Poseidon for UML - Professional	1.4	Sep-02	Java VM	\$699
Sybase	PowerDesigner	9	Dec-01	Windows	\$5990
Insoft	Prosa 2002 UML Modeler	5	Jun-02	Windows, Unix	\$1582
ProxySource	ProxyDesigner	1	Dec-00	Windows	\$0
Excel Software	QuickUML	1	Apr-01	Windows	\$495
Tri-Pacific Software	Rapid RMA	5.2	Nov-01	Java VM	n/a
Rational	Rational XDE Professional .NET Edition	2002	Feb-02	Windows	\$3595
Rational	Rational XDE Professional Java Edition	2002	Feb-02	Windows	\$3595
Artisan	Real-time Modeler	4.2	May-02	Windows	\$2495
Artisan	Real-time Studio Professional	4.2	May-02	Windows	n/a
I-Logix	Rhapsody Developer Edition	4	Sep-01	Windows	n/a

Rational	Rose Modeler	2002	Jan-02	Windows	\$1829
Rational	Rose Professional	2002	Jan-02	Windows	\$2442
Rational	Rose Enterprise	2002	Jan-02	Windows	\$4290
Rational	Rose Real Time	2002	Jan-02	Windows	n/a
Sodifrance	Scriptor	2.5	Oct-01	Java VM	\$4000
Magna Solutions	Silverrun Java Designer	1.1		Java VM	\$0
Adaptive Arts	Simply Objects Professional	3.3.3	Mar-02	Windows	\$800
Softera	SoftModeler Standard	3.6	Aug-02	Java VM	\$245
Softera	SoftModeler Professional	3.6	Aug-02	Java VM	\$495
Softera	SoftModeler Enterprise	3.6	Aug-02	Java VM	\$995
WebGain	StructureBuilder Enterprise	4.5.4	Jun-01	Java VM	\$4995
Popkin	System Architect	8.8	Aug-02	Windows	n/a
Telelogic	Tau UML Suite	4.3	Sep-01	Windows	n/a
TogetherSoft	Together Solo	6	Apr-02	Java VM	\$3495
TogetherSoft	Together Control Center	6	Apr-02	Java VM	\$5995
Pragsoft Corporation	UML Studio	6.2	Feb-02	Windows	\$500
Arion Software	UML2COM	1	Feb-01	Windows	\$990
UMLGraph	UMLGraph	1	Jul-02	Java VM	\$0
Umbrello	Umbrello UML Modeller	1.1	Oct-02	Linux	\$0
Unimodeler	Unimodeler	1.2	Aug-02	Linux	\$0
Visible Systems	Visible Analyst	7.6		Windows	n/a
Microsoft Visio	Visio 2002 Professional	2002	Mar-01	Windows	\$499
Visual Paradigm	Visual Paradigm for UML	1	Dec-01	Java VM	\$149
Visual Object Modelers	Visual UML Standard Edition	2.9.2	May-02	Windows	\$495
Visual Object Modelers	Visual UML Standard Edition for VB	2.9.2	May-02	Windows	\$795
Visual Object	Visual UML Plus	2.9.2	May-02	Windows	\$995

Modelers	Edition for VB				
Microsoft	Visual Studio .NET Enterprise Architect	1	Feb-02	Windows	\$2500
Excel Software	Win A&D Standard	3.4	Jul-02	Windows	\$495
Excel Software	Win A&D Desktop	3.4	Jul-02	Windows	\$1295
Microgold Software	WithClass Professional	2000	Mar-01	Windows	\$495
Microgold Software	WithClass Enterprise	2000	Mar-01	Windows	\$800
Tojosoft	d.vine	1	Apr-02	Delphi	\$69
Kennedy- Carter	iUML	2	Jan-01	Windows, Unix	\$3000
Mountfield Computers	mDesPro	5.5.2	Aug-02	Java VM	\$65

3.2 Servicio de construcción de sistemas

Los resultados del análisis del servicio de construcción de sistemas, dieron la pauta para el presente apartado. Algunas de las mejoras que a continuación se proponen, han sido extraídas de las memorias del Primer Seminario Internacional de Calidad en Desarrollo de Software, llevado a cabo en el auditorio del Instituto Tecnológico Autónomo de México (ITAM), campus Santa Teresa del 27 de mayo al 7 de junio del 2002.

Este seminario fue organizado por la Secretaría de Economía, el ITAM, la Asociación Mexicana de la Industria de Tecnologías de Información (AMITI) y Avantare Consultores y fue patrocinado por certum, la Secretaría de Economía, el Consejo Nacional de Ciencia y Tecnología (CONACYT), gedas México, IDS Comercial, DMR Consulting México, Grupo Nacional Provincial (GNP), el Instituto Mexicano del Petróleo (IMP), Microsoft, Itera e IBM. Se enmarcó dentro del Programa para el Desarrollo de la Industria del Software de la Secretaría de Economía y tuvo como objetivo promover y apoyar los esfuerzos por mejorar los procesos de desarrollo de software en México, así como difundir algunas de las alternativas más exitosas para desarrollar software de alta calidad.

Fue una buena noticia el notar, que muchos de los aspectos que se tocaron en este seminario, ya han sido implementados en certum obteniendo buenos resultados. El problema radica en que algunos de ellos incluso están documentados y no se siguen, y otros al contrario, se han aplicado y no están documentados. Para reducir la incertidumbre en la construcción de los sistemas, se cuenta entonces con dos propósitos:

1. Agregar al proceso oficial de construcción las mejoras encontradas
2. Implementar un método para asegurar que este proceso se ejecute en su totalidad en todos los proyectos. Esta implementación, compete, más que al servicio de construcción, a la planeación y control de proyectos, así que se hablará del mismo en el apartado siguiente (3.3 La planeación y control de proyectos).

Partiendo del actual diagrama de proceso del servicio de construcción de software (diagrama 3.2.1), a continuación se presentan los procedimientos que habrá de agregar para cumplir con el propósito no. 1, ilustrados por el diagrama 3.2.2.

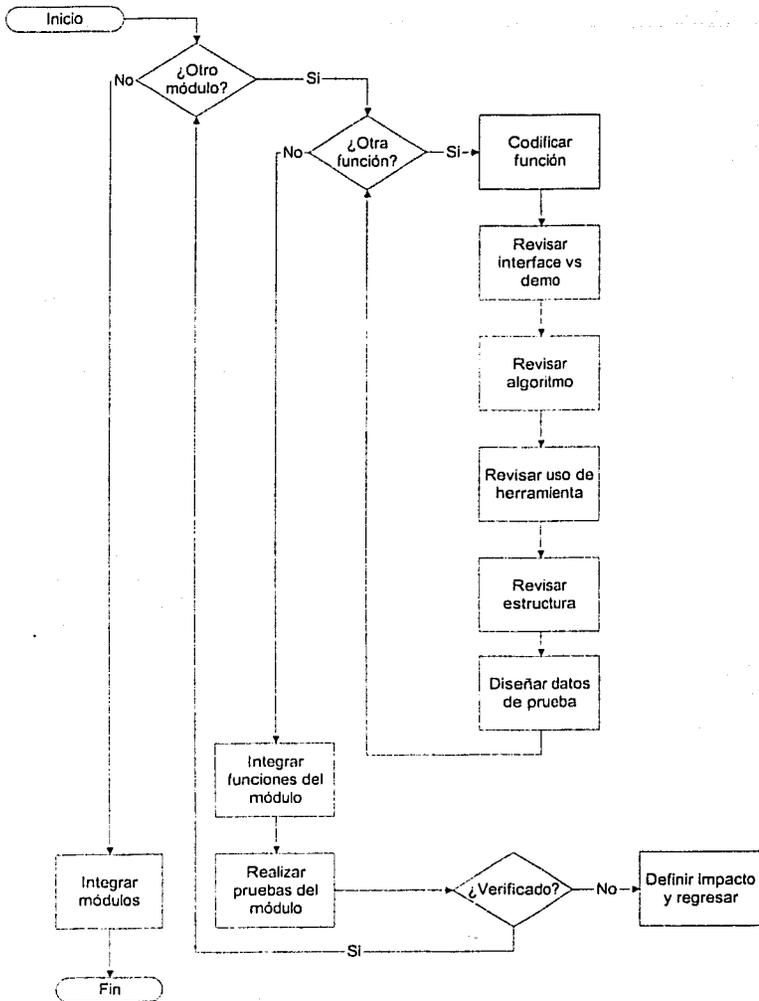


Diagrama 3.2.1 Diagrama actual del servicio de construcción

1. Inicio
2. Elaborar la ingeniería de detalle
3. ¿Procedimientos reusables?
Sí - 4
No - 5
4. Integrar procedimientos con librería general. Todo paquete o conjunto de pantallas que forman parte de un módulo o sistema, cuentan con un grupo de procedimientos y/o funciones de uso común con fines específicos para el paquete. Por ejemplo, si se trata de la recopilación de datos de contactos en una agenda, se contará con funciones para validar la captura de fechas, obtener la edad de una persona, el nombre de casada, nombre empezando por apellido, o información específica de acuerdo al sistema, p.ej. el puesto con mayor jerarquía con el que cuenta una persona, dado el acceso a datos del usuario.

5. Codificar función.

- Agregar consejos recomendables gracias a la experiencia obtenida durante la construcción, en una base de datos de asuntos (ver pag. 49). Esto constituye el "Libro de Conocimientos" (BOK - *Book of Knowledge*), que podrá ser consultado por el resto de los integrantes del equipo y por los que se integren en el futuro.
- Si se encuentran problemas que afecten el estimado, levantar un *asunto* con la pregunta correspondiente.
- Si el problema detiene por completo la construcción, pedir ayuda o convocar una junta técnica.

6. Agregar la nueva función a la matriz CRUD.

Una matriz CRUD, está constituida por el conjunto de funciones (organizadas en módulos) en las filas, y en las columnas se colocan los distintos objetos de la BD utilizados y/o alterados desde las funciones (tablas, vistas, secuencias, índices, etc.). En el interior de la matriz, se indican las letras:

C - Create
R - Read
U - Update
D - Delete

,según para lo que utilice cierta función, cada uno de los objetos.

El mantener una matriz como esta actualizada es de gran ayuda, pues antes de que uno de los desarrolladores modifique algún objeto de la base de datos (por lo general vistas y tablas), deberá consultar la matriz para verificar si alguna otra función la utiliza, y asegurarse de que sus cambios no afecten otros procedimientos. Este tipo de problemas es muy común si no se tiene una buena comunicación entre los desarrolladores, pero como no se puede confiar solamente en su buena memoria acerca de los objetos de la BD que utilizan, es bastante recomendable el uso de una matriz CRUD.

Actualmente existen productos que pueden generar reportes con este tipo de información, por ejemplo, Oracle Repository Objects® y Oracle Data Flow Diagrammer® que forman parte de la herramienta CASE de Oracle, Oracle Designer®.

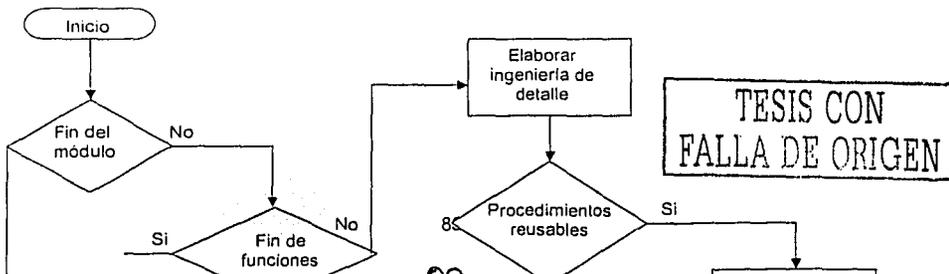
7. Verificar checklist de estándares de interfaz.
Si la función genera nuevos estándares o excepciones, agregarlos al checklist.
8. Verificar checklist de estándares de codificación.
Si la función genera nuevos estándares o excepciones, agregarlos al checklist.
9. Realizar pruebas unitarias (funcionales). Diseñar datos de prueba, si se requiere.
10. Revisión de código y uso de herramienta por parte de otra persona del equipo de construcción. Se levanta un incidente con las observaciones. Seguimiento del incidente (como se definió en la sección 1.5.2).

Este conjunto de pasos, sustituirían 6 de los actuales en el diagrama:

- Codificar función
- Revisar interfaz vs. demo
- Revisar algoritmo
- Revisar uso de herramienta
- Revisar estructura
- Diseñar datos de prueba

La "ingeniería de detalle" se refiere al diseño que realiza la persona que elaborará la función (pantalla) sobre cómo organizará el código para lograr los requerimientos deseados.

El objetivo es hacer que el ingeniero tenga una idea completa del funcionamiento a detalle de la pantalla que debe construir y sobre todo, cómo la construirá. Se recomienda contar con un esquema que pueda ser utilizado por cada uno de los desarrolladores en el que llene todos los datos requeridos. Algunos de estos datos pueden ser: nombre del archivo, procedimientos generales a utilizar, uso de variables globales, uso de procedimientos y/o funciones de base de datos, consultas principales a la base de datos, eventos a codificar y pseudocódigo de los procesos necesarios. El diseño de la construcción, previo a la codificación misma, reduce el riesgo de generar errores en el código, evita parches en los programas y promueve el reuso de procesos. La elaboración de la ingeniería de detalle está



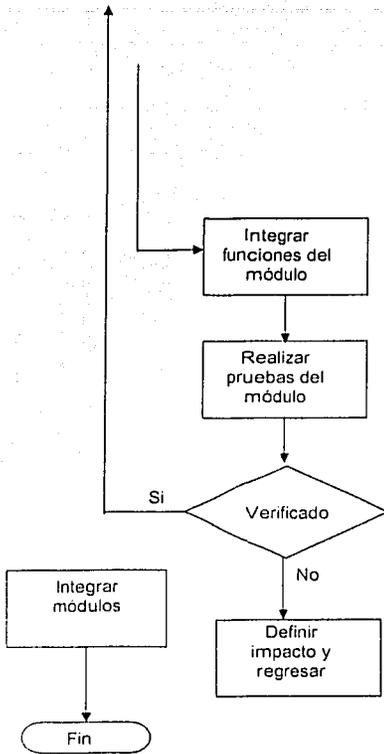


Diagrama 3.2.2 Diagrama mejorado del servicio de construcción

contemplada en diversos documentos con los que cuenta certum acerca de la construcción de sistemas, pero últimamente se ha olvidado y es tiempo de retomarlo.

Como ejercicio, se realizó la ingeniería de todas las nuevas pantallas que se construyeron para una nueva versión de un paquete y la cantidad de incidentes detectados en la fase de pruebas formales se redujo un

30%. Del total, el 70% fueron incidentes catalogados como 02 y 03.⁸ (ver tipos de incidentes en la sección 1.5.2)

Los procedimientos de revisión se encierran en uno solo, pues en realidad, cuando una persona revisa una función, toma en cuenta todos los aspectos que la conforman, es decir, el algoritmo y el uso de herramienta. La revisión de estructura está implícita, y se ha colocado la revisión de interfaz vs. demo en manos del programador, pues no se requiere de una persona con mayor experiencia para hacerlo.

A simple vista, pareciera que lejos de ayudar, este proceso complica aún más la construcción, pero se ha comprobado que la aplicación de cada uno de los puntos es de mucha utilidad, pues reduce el número de incidentes detectados en la fase de pruebas formales, y así, la incertidumbre en la calidad. Cabe resaltar que mientras más pronto se encuentren los errores y se pida ayuda en caso de requerirla, se gana más tiempo en las fases de integración de las funciones y módulos del sistema.

TESIS CON
FALLA DE ORIGEN

⁸ Fuente: Resultados de pruebas formales para la cuenta más importante en certum, 2003

3.3 La planeación y control de proyectos

El análisis realizado en la segunda parte de esta investigación, sobre la planeación y control de proyectos llevado a cabo por certum, dio como resultado que, dentro de la empresa, están implementados los puntos más importantes de la planeación formal y de la planeación estratégica. Se resaltó el uso de las estrategias del reporte diario de actividades y el conteo de puntos por función como distintivos de la organización y herramientas clave para reducir la incertidumbre y cumplir así con la política de calidad.

Todo este esfuerzo dio buenos resultados en el pasado y deja mucho que desear en el presente, pues los resultados financieros han reflejado pérdidas y los productos finales han reportado muchos errores de funcionalidad, no se han entregado en el tiempo establecido y han resultado con más presupuesto del estimado. Es tiempo entonces, de no sólo preocuparse por las técnicas de diseño y construcción de sistemas, sino de adoptar una metodología de planeación específica del área de desarrollo de software, abarcando también elementos organizacionales y estructuras que se necesitan para mejorar todo el proceso.

Por supuesto que existen proyectos excepcionales en los que se ha tenido la fortuna de contar con personas en los roles de líder de producto y aseguramiento de la calidad, que ponen su mayor esfuerzo en aplicar al pie de la letra el proceso de planeación y control, y que además le agregan un valor adicional de mucha ayuda, que es el análisis de los logros y fracasos del proyecto; resultados que sirven de alimentación para la planeación de siguientes versiones, e incluso de nuevos proyectos.

Con base en este tipo de proyectos exitosos y en las técnicas extraídas del Primer Seminario Internacional de Calidad en Desarrollo de Software, se presentarán algunas alternativas de implementación de procesos de sistemas informáticos. Cabe mencionar, que este seminario estuvo conformado por cinco tutoriales y cinco talleres impartidos por el profesor Pankaj Jalote, Jefe del Departamento de Ciencias de la Computación e Ingeniería del Instituto de Tecnología de la India (en Kanpur), así como por un conjunto de conferencias de diversos temas relacionados con las mejores prácticas mundiales de Administración de Proyectos e Ingeniería de Software dictadas por miembros de las instituciones organizadoras, expertos mexicanos de la industria y Daniel Roy, Presidente de Software Technology, Process & People, Inc.

Se comenzará, presentando la diferenciación que hace el profesor Jalote entre los procesos "maduros" y los "inmaduros".

3.3.1 Procesos maduros e inmaduros

Las afirmaciones siguientes son aplicables a todo tipo de proceso diseñado e implementado en una organización para definir los estándares de trabajo y el orden de las tareas a realizar para lograr la conclusión de un proyecto o de una parte de un proyecto.

CARACTERÍSTICAS DE LOS PROCESOS INMADUROS

1. Proceso improvisado *ad hoc*³ por los participantes
2. El éxito depende mayormente de las personas involucradas
3. Comunes problemas de costos y agenda
4. La funcionalidad y calidad del producto es comúnmente castigada para respetar la agenda
5. No es posible implementar nuevas tecnologías por falta de organización
6. La calidad del producto es difícil de predecir

CARACTERÍSTICAS DE LOS PROCESOS MADUROS

1. Procesos definidos, documentados y en mejora continua
2. Son comprendidos y aplicados por todos los participantes
3. Probabilidades de éxito poco dependientes de las personas
4. Los departamentos de mayor jerarquía proporcionan visibilidad en el proceso
5. Seguimiento de procesos bien controlado
6. Uso disciplinado de nuevas tecnologías

Desgraciadamente, certum cuenta con más características del primer grupo que del segundo, en especial con la característica número 2. Es muy común que aún contando con procesos definidos y documentados, no están en mejora continua si no es por pocas personas que muestran el interés por hacerlo. Se han producido proyectos exitosos gracias a que sus participantes han sido seleccionados minuciosamente entre el personal y nuevas contrataciones y han sido su esmero, conocimiento y experiencia lo que lo ha hecho posible. El éxito debe estar basado en el seguimiento fiel de procesos bien diseñados, reduciendo así el riesgo. Gracias a que certum ha pertenecido al grupo de procesos maduros anteriormente, cuenta con buenas posibilidades de modificar su situación actual. Solo hace

³ Expresión latina que significa: "lo que se dice o hace sólo para un fin determinado"

falta renovarse e implementar un programa de transformación de sus procesos para hacerlos "maduros" y mostrar a todos sus colaboradores las mejoras, además de la gran importancia que tiene el apegarse a los procesos al pie de la letra. Para profundizar en este tema se presenta a continuación una reseña sobre el aseguramiento de calidad de software.

3.3.2 Aseguramiento de Calidad de Software (SQA – Software Quality Assurance)

El ser humano tiene como naturaleza la tendencia a tomar atajos y a la resistencia a seguir procesos definidos. *certum* cuenta con la experiencia de fracasos originados por seguir estas tendencias, por lo que requiere de la implementación de un programa de SQA.

La calidad de software se diferencia del "control de calidad", en que el segundo pone su atención en la calidad del producto final. Lo que compete a la calidad de software es la calidad de los procesos; también es denominado como "auditorias de software". Estas auditorias no deben ser realizadas por personas que formen parte del proyecto, sino de un grupo especial de SQA¹⁰.

CARACTERÍSTICAS DEL GRUPO DE SQA

1. Cualquier persona entrenada que sea externa al proyecto, puede formar parte del grupo
2. Debe difundir el proceso tanto dentro del grupo como a los involucrados en el proyecto
3. Requiere de un coordinador de calidad de software entrenado en auditorias
4. Desarrolla estrategias, áreas críticas y frecuencia de las auditorias
5. Programa las auditorias regularmente y asegura que se cubran las áreas del proceso en su totalidad
6. Construye herramientas de apoyo como listas de verificación, de manera que su trabajo se realice de manera impersonal y honesta
7. Cuenta con métodos formales para reportar los resultados y provocar que el proyecto se desarrolle según los estándares definidos en los procesos

El llevar a cabo estas sencillas tareas, puede originar grandes resultados, pues todo se resume a contar con disciplina en la ejecución de las tareas, documentar cada paso y utilizar los resultados como datos de entrada para nuevos proyectos.

¹⁰ JALOTE, Pankaj "Software Project Management in Practice", Ed. Addison Wesley, EUA 2002

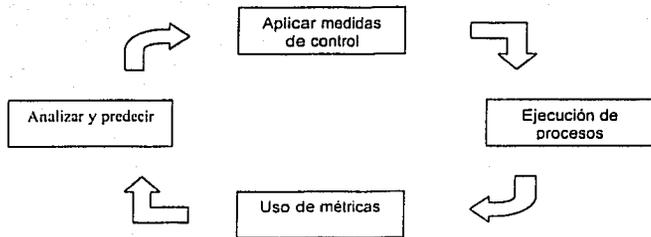
Se pueden implementar estas auditorias no solamente en los niveles más altos de los procesos, como son el diseño y construcción de software, sino llevarlo también a otras tareas que son sumamente importantes, como la estimación de costos, la creación del panel de control, el seguimiento de incidentes, la aplicación de métricas etc. pues son algunas de las técnicas que actualmente están documentadas pero que es fácil que no se lleven a cabo tal y como fueron diseñadas.

3.3.3 Monitoreo y control de proyectos de software

Anteriormente se ha recalcado la importancia de llevar un adecuado control de los proyectos, de modo que se puedan detectar los riesgos oportunamente y sean atacados antes de que las consecuencias sean irreversibles. La mejor manera para monitorear un proyecto es especificando las expectativas y el estatus cuantitativamente. Existen dos aspectos clave:

1. Obtener información acerca del estado actual del proyecto
2. Aplicar acciones correctivas si el estado es "no saludable"

El monitoreo y el control van de la mano. Cuando no se requiere de control, la información acumulada durante el monitoreo lo demuestra. El objetivo principal de estas actividades es asegurar que el proyecto alcance todas sus metas y se da con el seguimiento de los siguientes puntos:



CARACTERÍSTICAS PRINCIPALES DEL MONITOREO Y CONTROL

1. Se realiza entre los mismos compañeros del equipo
2. Es un proceso estructurado con roles definidos para cada uno de los participantes
3. La atención se centra en la identificación de problemas, no en su solución
4. La información obtenida es almacenada y utilizada para analizar la efectividad del monitoreo
5. Por lo general, es difícil convencer a la gente para que lo use, pero puede impactar a toda la organización
6. Si el proceso está bien documentado, la implementación es sencilla
7. Requiere de entrenamiento y del apoyo de listas de verificación
8. Contar con experiencia en recolección de información puede ayudar en optimizar métodos y lineamientos
9. Requiere de mucha "evangelización"

Conclusiones

En este tercer y último capítulo, se propone la implementación de 3 cambios que afectan de manera estructural los procesos de diseño y construcción de sistemas, así como el proceso de control de proyectos. No es tan desconcertante ni tan difícil de implementar dado que el giro que se propone no es tan pronunciado, sólo requiere de más disciplina y atención en pequeños detalles que pueden significar grandes resultados.

Acerca del cambio en el diseño, la sección 3.1 sugiere la utilización de una herramienta flexible, genérica y sencilla como lo es el UML, que promete además de la estandarización de los modelos lógicos en toda la empresa, el control máximo de los requerimientos presentándolos de la manera más transparente y comprensible tanto al equipo de trabajo como al cliente. Además, constituye una herramienta muy útil para la fase de construcción del sistema, pues ayuda a tener una visión global de lo que se requiere codificar, permitiendo la detección de procesos reusables. Esto, junto con un proceso organizado de construcción que contemple la documentación y pruebas necesarias, como el que se indica en el apartado 3.2, debe realizarse bajo una inspección y control minuciosos que garanticen que todo el equipo trabaja al mismo ritmo y avanzando en dirección a los mismos objetivos. Dicho control puede ser implementado llevando a cabo las prácticas definidas en la sección 3.3.

Lo único que resta es la visión de los posibles escenarios a los que se pueden enfrentar las empresas mexicanas proveedoras de sistemas de software a la medida. Para definir dichos escenarios se seguirá el comportamiento de dos **variables fundamentales**:

1. La implementación de tecnología de software.

A lo largo de la presente investigación, se ha recalado la importancia de mantenerse actualizado en este aspecto, tomando en cuenta las áreas de diseño, construcción, calidad, costeo, planeación y control de proyectos de software. Todas ellas van evolucionando con el paso del tiempo, por lo que su implementación es un proceso continuo, mejorando internamente los procesos y manteniéndose informados de las innovaciones que ofrecen los proveedores. Por ejemplo, las mejoras aquí propuestas, pueden significar para certum una gran reducción de la incertidumbre en la calidad, tiempos de entrega y costos, que implementadas correctamente, se podrían obtener resultados visibles y medibles dentro de 3 ó 5 años, pero lo que puede llegar a ser el diferenciador de una empresa, es la constante actualización de tecnología de software que permita ofrecer un alto grado de competencia, tanto a nivel nacional como internacional.

2. Nivel de los ingenieros

Todos los recursos son importantes, pero sin duda alguna, los recursos humanos constituyen el motor de toda empresa, especialmente de aquéllas en las que la fuerza de trabajo no está basada en máquinas ni combustibles. Todo el poder de una fábrica de software está basado en el rendimiento y eficiencia de los ingenieros, así se trate de programadores, diseñadores, líderes de producto o de proyecto. Cabe aclarar que se les denomina "ingenieros" debido al concepto adoptado por *certum*, sin importar si se trata de licenciados en informática, actuarios, ingenieros en sistemas, licenciados en ciencias de la computación o en matemáticas aplicadas y computación, etc.

La fluctuación de las variables anteriores provoca distintos comportamientos, mismos que pueden ser observados y analizados de acuerdo con la posición que adopten las siguientes **variables secundarias**:

1. Competitividad nacional e internacional.

El correcto equilibrio entre las dos variables fundamentales puede propiciar la competitividad, obligando a las empresas a ofrecer a sus clientes cada vez menor incertidumbre en el costo de los productos, la fecha de entrega y calidad de los mismos. Al mantener la competencia alta dentro del país, se puede favorecer la situación a nivel internacional y lograr que tanto empresas nacionales como extranjeras prefieran los servicios de sistemas a la medida que ofrecen proveedores mexicanos.

2. Personal certificado.

Dentro del área informática, se va ha vuelto muy importante el que el personal no sólo esté capacitado para manejar herramientas de diseño y construcción de software, administrar bases de datos y sistemas operativos, sino que además se encuentre certificado por las firmas que generan y distribuyen este tipo de herramientas. En la medida en que una fábrica de software cuente con personal certificado en el uso de las herramientas que utiliza, se contará con un buen método de venta y aseguramiento de la calidad de los productos, y así aumentará la confiabilidad de los clientes en los mismos.

3. Compensaciones justas.

El método de compensaciones de personal es un factor que define la permanencia o la fuga de cerebros, no solo de empresa a empresa, sino de país a país. De nada sirve fomentar el crecimiento del nivel de los ingenieros para que implementen lo último en tecnología de desarrollo de software, si no cuentan con la remuneración que otras organizaciones les puedan ofrecer por hacer el mismo trabajo. Este es solo uno de los muchos detalles que la administración debe cuidar para asegurar la existencia de la empresa en el futuro. El porcentaje manejado en los escenarios siguientes se refiere

al número de empresas mexicanas que logran ofrecer salarios justos a sus ingenieros por los servicios que les prestan.

Escenario "La vida es bella", 15% de probabilidad de que ocurra

Los riesgos son vislumbrados y atacados con anticipación, el ambiente de trabajo es bueno y la productividad alta. Los buenos resultados facilitan la reventa y la captación de nuevos clientes y nuevos retos, lo que impulsa el avance continuo de las personas involucradas en el desarrollo de los sistemas.

Las condiciones para que se cumpla este escenario son:

- i. Los estudiantes de licenciaturas relacionadas con el área de la informática egresan con conocimientos fuertes sobre diseño computacional y uso de herramientas actuales para construcción de software.
- ii. Existencia de vastos programas de becarios que inducen a los recién egresados de universidades al mundo laboral, de modo que permitan la explotación de los conocimientos aprendidos.
- iii. Los continuos conflictos en Estados Unidos por su lucha contra el terrorismo causan cada vez más polémica en el mundo. Las inversiones lejos de aumentar, disminuyen por la inestabilidad de su economía, lo que provoca que los grandes corporativos del extranjero prefieran contar con proveedores de tecnología de información pacifistas. Tal es el caso de México, que por su cercanía a la mencionada potencia mundial y por sus notables avances en el ramo cada vez capta mayor atención por parte de inversionistas y clientes.
- iv. Las empresas constructoras de software a la medida apoyan a su personal para que obtengan la certificación en las herramientas del área de su preferencia. Dicho apoyo es muy bien aprovechado por ingenieros de todos los niveles; se preparan continuamente para obtener la certificación y se mantienen informados sobre los requisitos y trámites necesarios.
- v. Las empresas mexicanas no solo se mantienen al día en cuanto al desarrollo de técnicas y herramientas de tecnología de software, sino que la investigación recibe cada vez más apoyo y México se aproxima a formar parte de los países exportadores de tecnología.

Escenario "Rápidos y Furiosos", 40% de probabilidad de que ocurra

Surge la analogía con el título "Rápidos y Furiosos", pues en pocos años se logra contar con personal altamente capacitado pero no su fidelidad, entorpeciendo así los objetivos de las empresas mexicanas de mejora constante en busca de la competitividad internacional. Aún así se logran captar algunos clientes que con problemas, se conservan. Condiciones para que se presente:

- i. Los estudiantes de licenciaturas relacionadas con el área de la informática egresan con conocimientos fuertes sobre diseño computacional y uso de herramientas actuales para construcción de software.
- ii. Existencia de vastos programas de becarios que inducen a los recién egresados de universidades al mundo laboral, de modo que permitan la explotación de los conocimientos aprendidos.
- iii. Nuestro país vecino de la frontera norte se consolida como potencia mundial. Sus ataques en regiones del medio oriente aunque siguen sin ser aceptadas desde el punto de vista moral por el uso de la fuerza, pero logran el respeto de países que se unen a sus propósitos al notar su crecimiento económico. Prolifera la industria y la ubicación de corporativos que ofrecen software a la medida en territorio de países como México, que cuentan con profesionistas capacitados.
- vi. Las empresas mexicanas luchan por mantener el ritmo de la competencia extranjera. Continuamente tratan de actualizar las versiones de software, y equipo de hardware. Se cuenta con un departamento destinado a buscar, evaluar e implementar técnicas de diseño, planeación y control de proyectos de construcción de sistemas. La competencia es dura y la fuga de cerebros alta. Son pocas las empresas que logran ser competitivas y necesitan cada vez más de la asociación con capital estadounidense.

Escenario "Camino a la perdición", 15 % de probabilidad de que ocurra

El título lo dice todo, sin personal capacitado ni el esfuerzo por implementar nuevas tecnologías, no se cuenta con otro camino mas que "el de la perdición". La incertidumbre en la economía mexicana no se detiene, efecto que causa inseguridad para invertir. Las condiciones para que se presente son:

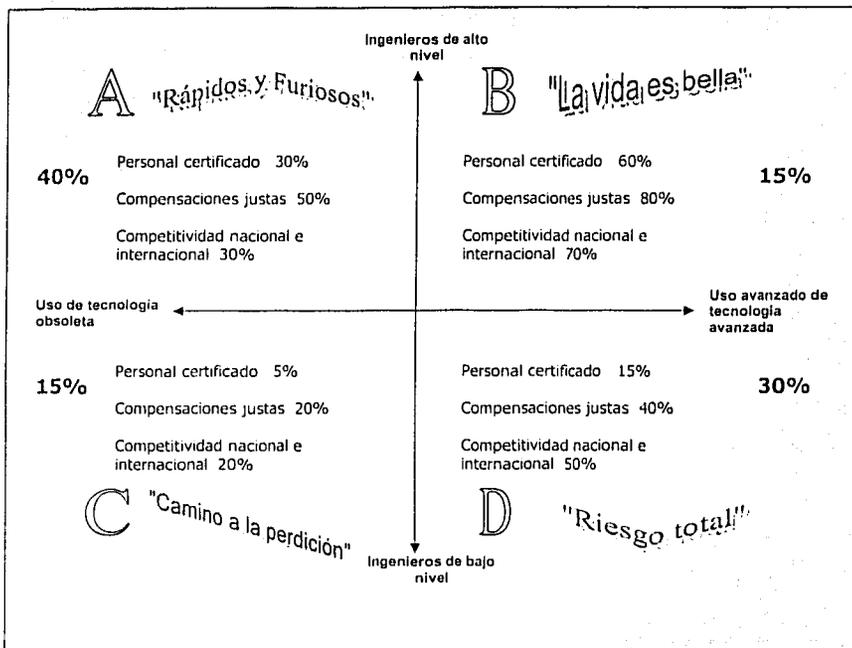
- i. La docencia es descuidada en las universidades, los estudiantes se encuentran poco motivados debido a las bajas expectativas de empleo que el país ofrece.
- ii. El recién egresado se coloca con dificultad en la industria de sistemas debido a los requisitos de experiencia. Falta de actitud emprendedora para el inicio de un negocio propio, donde una de las causas es la desconfianza económica. Los factores anteriores se convierten en causas de subempleo.
- iii. En la industria mexicana, el tipo de proyectos no promueve el desarrollo del personal. Caso omiso a las tendencias y el avance de la ingeniería de software
- iv. Falta de programas que impulsen la certificación en herramientas de tecnología de software. Hay poco interés y aptitudes para llevar a cabo auto-capacitación para mantenerse actualizados; el acelerado avance de la tecnología es visible, así que los conocimientos se vuelven obsoletos rápidamente.
- v. México no es considerado candidato por parte de empresas líderes en el ramo para la colocación de industrias y sucursales de proveedores de sistemas a la medida.

Escenario "Riesgo total", 30% de probabilidad de que ocurra

Este escenario puede ser considerado como una fase anterior al de "Camino a la perdición", pues si no se detectan los riesgos oportunamente, la mala administración no permite contar con capital de apoyo para sostenerse y mucho menos para invertir y recompensar a los buenos elementos, poco a poco las empresas de sistemas a la medida irán descendiendo. Sin embargo, si los riesgos son atacados y se implementan procesos disciplinadamente que impulsen la mejora continua, podría esta no ser una fase anterior a la "perdición", sino a "Rápidos y Furiosos". Las condiciones para que se cumpla son:

- i. La docencia es descuidada en las universidades, los estudiantes se encuentran poco motivados debido a las bajas expectativas de empleo que el país ofrece.
- ii. El recién egresado se coloca con dificultad en la industria de sistemas debido a los requisitos de experiencia. Falta de actitud emprendedora para el inicio de un negocio propio, donde una de las causas es la desconfianza económica. Los factores anteriores se convierten en causas de subempleo.
- iii. Las pocas empresas mexicanas consolidadas en la construcción de software a la medida continúan sus esfuerzos por el uso de técnicas avanzadas de desarrollo de sistemas, pero solo son malas copias de implementaciones logradas por la competencia

- iv. En el extranjero, los líderes en el ramo desvían su mirada y sus inversiones de México, prefiriendo países que les ofrezcan menor incertidumbre en su economía, una mejor cultura de negocio y estabilidad social.



Escenarios para el año 2010¹¹

El diagrama anterior permite vislumbrar los distintos escenarios para el año 2010 generados a través de las variables mencionadas y se describen a continuación mediante la explicación del comportamiento de las variables secundarias.

¹¹ Creación propia

"Rápidos y Furiosos", 40% de probabilidad de que ocurra.

Personal certificado	Comparte las mismas expectativas de nivel del personal que el escenario B. Sin embargo, el personal calificado se reduce a la mitad. Algunas personas que han logrado certificarse encuentran mejores ofertas de trabajo en empresas extranjeras.
Compensaciones justas	La falta de competitividad provocada por la escasa actualización en tecnologías de construcción de sistemas, no permite a la organización igualar los salarios a los de empresas extranjeras, pero se logra un equilibrio entre las nacionales fuertes, manteniendo a gran parte de su personal tranquilo, por contar con un ingreso por arriba del promedio ejerciendo su profesión.
Competitividad nacional e internacional	Se ha intentado promover el uso de técnicas avanzadas para el desarrollo de sistemas, pero la fuga del personal capacitado dificulta, y en ocasiones detiene su correcta implementación. Sin embargo, algunas empresas se sostienen cuidando a su personal calificado y tratando de impulsar al resto, esperando así lograr cada vez un mejor diseño y seguimiento de los procesos para el desarrollo de software que resulta en un aceptable nivel de competitividad.

"La vida es bella", 15 % de probabilidad de que ocurra.

Personal certificado	El 60% del personal de las fábricas mexicanas de software cuenta con algún tipo de certificación, lo que reduce en alto grado la incertidumbre de los productos y mantiene las ventas constantes.
Compensaciones justas	El 80% de las empresas es capaz de mantener salarios justos por el alto nivel de sus trabajadores, los cuales se mantienen en México impulsando la industria del desarrollo de sistemas en lugar de aceptar ofertas extranjeras.
Competitividad nacional e internacional	La competitividad nacional crece aceleradamente propiciando productos de mejor calidad con el costo y fecha de entrega estipulado, factores que ayudan a las empresas mexicanas a competir con firmas internacionales y se logra abarcar el mercado mexicano e incluso algunos proyectos para empresas en el extranjero, implementando sistemas gracias al uso avanzado de tecnología de punta.

"Camino a la pérdida", 15% de probabilidad de que ocurra.

Personal certificado	Son muy pocos los ingenieros que logran una certificación oficial, en incluso los que la tienen, prefieren ejercer su profesión en otro país, pues la situación en México no es nada prometedora para poder asegurar un patrimonio, ni siquiera para profesionistas con postgrados.
Compensaciones justas	La falta de inversión y la fluctuación incesante de la moneda, impide a las empresas compensar como se debe a sus trabajadores. La inflación se mantiene por encima de los salarios, las prestaciones cada vez son menores y no se vislumbran tiempos mejores. Incluso los

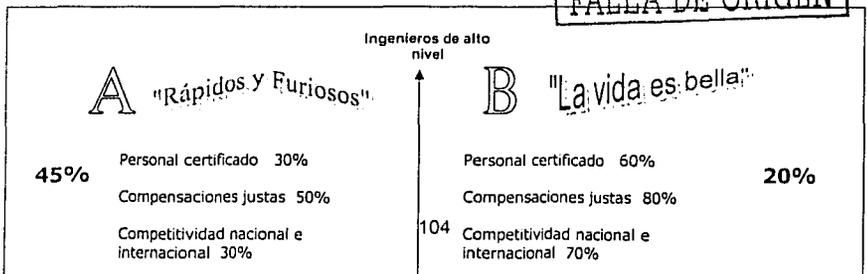
	ejecutivos, directivos y socios han resentido el descenso de su salario ante la economía nacional y son obligados a administrar mejor sus bienes.
Competitividad nacional e internacional	No hay competencia por calidad. Las empresas se ven en la necesidad de competir por costos sin ser capaces de mostrar transparencia a sus clientes sobre los procesos. Los proyectos grandes prefieren a proveedores de sistemas extranjeros, dejando a los mexicanos sencillos sistemas de consultas, reportes y mantenimientos. Las que se han sostenido por encima de la mayoría lo han logrado con grandes esfuerzos, reduciendo su nómina y los precios por sus servicios. El "camino a la perdición" es amplio y se deja ver el acaparamiento del mercado por parte de empresas extranjeras que ni siquiera cuentan con oficinas en territorio mexicano.

"Riesgo total", 30% de probabilidad de que ocurra.

Personal certificado	El bajo nivel con el que salen los egresados de las universidades y la falta de apoyo por parte de las empresas para aceptar gente sin experiencia y además negándose a pagar por la capacitación y exámenes de certificación, impide el desarrollo de profesionistas del área de sistemas computacionales. Solo quienes cuentan con antigüedad en las empresas logran certificarse, en su mayoría por su propia cuenta gracias a la experiencia adquirida en el día a día.
Compensaciones justas	Dado el progreso que se ha logrado en la actualización de tecnología de software, las empresas podrían compensar mejor a sus empleados, pero por errores en la toma de decisiones y administración, los ingresos disponibles no son suficientes para que los salarios superen a la inflación.
Competitividad nacional e internacional	La competitividad internacional es escasa, existen algunas excepciones por parte de pocas empresas que han logrado sostener clientes en el extranjero. En el interior, la gran mayoría se encuentran compitiendo por costo y no por calidad, en ocasiones consiguen proyectos en las que hay que involucrar hasta 20 ingenieros construyendo pero los resultados dejan mucho que desear.

Siguiendo el curso de las condiciones descritas, se presentan ahora los escenarios para el año 2020:

TESIS CON
FALLA DE ORIGEN





15%	Personal certificado 5% Compensaciones justas 20% Competitividad nacional e internacional 20%	Personal certificado 15% Compensaciones justas 40% Competitividad nacional e internacional 50%	20%
------------	---	--	------------

C "Camino a la perdición" D "Riesgo total!"
 Ingenieros de bajo nivel

Escenarios para el año 2020 ¹²

¹² Creación propia

TESIS CON
FALLA DE ORIGEN

"La vida es bella", 20% de probabilidad de que ocurra

Personal certificado	El número de organismos que cuentan con las licencias necesarias para aplicar exámenes oficiales de certificaciones ha crecido, lo que impulsa el reconocimiento de desarrolladores, DBA's (<i>DataBase Administrators</i>), encargados de redes y sistemas operativos por parte de las firmas proveedoras de tecnología.
Compensaciones justas	El país ha logrado sostener su economía e incluso reactivarla después de momentos difíciles, lo que favorece las inversiones nacionales y extranjeras y así, la rentabilidad de las empresas en general. Las proveedoras de sistemas a la medida no son la excepción, por lo que se declaran aptas para ofrecer salarios competitivos a sus colaboradores.
Competitividad nacional e internacional	Los esfuerzos realizados por empresas mexicanas para mejorar y aplicar procesos específicos de cada tarea dentro del desarrollo de sistemas a la medida han rendido frutos. Su productividad puede ser comparada con la de compañías extranjeras que habían sido reconocidas como las mejores en el ramo.

"Camino a la perdición", 15% de probabilidad de que ocurra

Personal certificado	La cantidad de personas mexicanas certificadas en el uso de alguna herramienta es mínima. El poco interés es derivado de que las empresas del área de sistemas, e incluso su personal son en gran mayoría de origen extranjero.
Compensaciones justas	Para los profesionistas extranjeros los empleos como analista, programador, líder de producto, investigación tecnológica, etc., son muy bien remunerados tanto en México como en sus propios países. Desgraciadamente, son muy raros los casos en los que mexicanos llegan a ocupar puestos importantes por el escaso conocimiento de nuevas tecnologías y por no contar con alguna certificación, a menos que cuenten con capacitación en el extranjero.
Competitividad nacional e internacional	Las empresas mexicanas dedicadas al desarrollo de grandes aplicaciones a la medida han desaparecido. No pudieron seguir compitiendo contra las grandes compañías hindúes y estadounidenses, quienes se han ocupado de aplicar procesos avanzados de desarrollo y seguimiento de procesos. Las pocas y pequeñas fábricas de software mexicanas se ocupan de sistemas menores.

"Rápidos y furiosos", 45% de probabilidad de que ocurra

Personal certificado	El porcentaje de certificaciones en México en el área de sistemas ha aumentado, incluyendo a aquellos que han tenido la oportunidad de
----------------------	--

	capacitarse en el extranjero. Aunque una parte de estas personas han preferido laborar fuera del país, son más los que deciden quedarse para iniciar incluso su propio negocio a pesar del rezago tecnológico.
Compensaciones justas	Si bien han proliferado las compañías nacionales, las compensaciones no son comparables a las que se ofrecen fuera del país, aunque suficientes para mantener una vida holgada. El dedicarse a la industria de sistemas de información sigue siendo una buena opción, en la que el tratar de mantenerse actualizado en cuanto a los últimos avances tecnológicos es la clave para continuar el desarrollo tanto profesional como social.
Competitividad nacional e internacional	Al comparar el tamaño de los proyectos y las sumas de dinero negociadas internacionalmente, las compañías mexicanas siguen quedando atrás. Sin embargo, en el interior existe una competencia sana que obliga a mejorar continuamente mediante el seguimiento de procesos y calidad de software. Aún se trata de seguir los pasos de las empresas sostenidas y triunfadoras, que en ocasiones proporciona buenos resultados, pero México sigue sin ser reconocido como un país que impulse la ingeniería de software.

Conclusiones generales

En lo que se refiere a sistemas de información, el contar con el software no es suficiente si éste no cumple con todas las expectativas y nivel de operación de la empresa, por lo que una adecuada planeación bajo la cual se rija el proceso de diseño y construcción de sistemas, así como tener conocimiento de las técnicas de diseño y herramientas de construcción necesarias, son piezas clave para que el sistema sea un éxito, al cumplir con las expectativas de calidad, costos y fechas de entrega.

De acuerdo al objetivo planteado para esta investigación, se realizó un análisis de la efectividad de las técnicas utilizadas para desarrollar y controlar los proyectos de diseño y construcción de aplicaciones a la medida. A partir de dicho análisis, se pudo determinar que certum no ha hecho caso omiso de las recomendaciones de las empresas internacionales exitosas que impulsan a la utilización de herramientas de planeación como piezas clave para dirigir la organización, no solo en su área administrativa o financiera, sino aplicándolas también en los procesos de producción.

Se pudo llegar a esta conclusión después de encontrar la aplicación de muchos de los aspectos importantes de la planeación así como de técnicas de ingeniería de software en certum, entre los cuales se encuentran, por mencionar algunos:

- La asignación de actividades a cada uno de los miembros del equipo se realiza tomando en cuenta habilidades, pretensiones de desarrollo y tiempos, como se hizo notar en la sección 2.1.3
- El apartado 2.1.3 también refleja que al determinar el tiempo disponible no se olvidan los tiempos para pruebas, revisiones, correcciones e imprevistos originados por falta de conocimiento de las herramientas de desarrollo (curva de aprendizaje), problemas técnicos en el manejo del hardware o funcionamiento de las conexiones en red, etc.
- Se encuentran claramente definidos los pasos a seguir para llevar a cabo el control del plan, definiendo los puntos de control en lugares estratégicos, susceptibles de cambios de acuerdo al avance del proyecto. Dichos puntos de control se encuentran enumerados en la sección 1.5.1.
- Gracias a la información explotada a través del sistema SILPRO, detallado en el apartado 1.4, el avance se estudia junto con el encargado del producto cada determinado tiempo según el tamaño del proyecto, por lo general una vez a la semana, aplicando eficientes técnicas de medición de software (conteo de puntos por función – ver pag. 47)
- Todas las personas involucradas participan en la planeación, estimando tiempos y aportando ideas aplicables al diseño y a la metodología de construcción, situación que se expone en la pag.

57

Sin embargo, observando con detenimiento el proceso real que se sigue durante el desarrollo de software, se encontró que certum ha caído en la tendencia de tomar caminos cortos ignorando en ocasiones algunos puntos incluidos en el proceso. Esto es el origen de la incertidumbre en la calidad, y lo que provocó el resultado de la evaluación interna SPICE realizada por el director general, el cual ubicó a certum en el nivel 2 de productividad. Se acudió entonces al estudio de los lineamientos propuestos por el profesor Pankaj Jalote, quien es el responsable de haber llevado a empresas hindúes de desarrollo de software a los niveles máximos de productividad (por ejemplo, "Infosys"). El último apartado de esta investigación puede resultar de gran utilidad para empresas pequeñas, medianas y grandes dedicadas a la tecnología de información, pues los datos son actuales y las técnicas están siendo implementadas obteniendo grandes beneficios como es el caso de las empresas de la India. Constituye además una invitación a continuar investigando sobre los estudios del profesor Jalote y sus colaboradores a través de la bibliografía consultada, pues contiene métodos y técnicas para aquéllos desarrolladores que estén iniciando y deseen contar con una buena planeación de sus procesos. Este no es el caso de certum, por lo que en la presente investigación se hizo hincapié en las técnicas de aseguramiento de la calidad y en el monitoreo y control de proyectos.

Para cumplir con los tiempos de entrega, y reducir la diferencia entre los costos reales y los pronosticados inicialmente, se sugiere renovar las técnicas de diseño y construcción de sistemas. Esta propuesta se basa principalmente en la implementación del Lenguaje de Modelado Unificado (UML), por la gran aceptación que ha tenido a escala mundial en el área de diseño de bases de datos. La cantidad y calidad de información acerca del tema crece día con día, encontrando bibliografía especializada de acuerdo a distintas herramientas de desarrollo como Visual Basic. La página oficial en Internet (<http://www.rational.com/uml>) puede ser de mucha utilidad para personas que buscan desde cómo comenzar, hasta aquéllas que requieren de soluciones específicas para casos reales, encontrando ejemplos y sugerencias de software para aplicar la ideología del UML a todo tipo de requerimientos. En cuanto a la construcción de los sistemas, se determinó que lo establecido en el Diseño de Servicios® ha quedado obsoleto y es rara vez consultado. Se propone un nuevo plan de trabajo, que contenga los aspectos generales a contemplar en la construcción de todo sistema, tomando en cuenta el cuidado minucioso de la documentación que registre la organización del código y que permita un sencillo mantenimiento en lo posterior.

Se espera que con la aplicación de las mejoras incluidas en esta investigación, se logre reducir la incertidumbre en la calidad, fechas de entrega y costos, de manera que sea posible notar cambios satisfactorios reconocidos por los clientes en un rango de 3 a 5 años, pero sobre todo, que la

productividad mejore y se mantenga por muchos años más, y así, que la empresa logre consolidarse dentro de las líderes en el ramo a nivel internacional.

El presente trabajo constituye entonces, una propuesta para recordar los inicios de la compañía ("Back To Beginnings - BTB") y la época en la que se comenzaron a diseñar los estándares de operación y procesos de planeación. Es el primer paso para comenzar a revisar la historia y hacer una reestructuración haciendo uso de lo último en diseño, construcción y control de aplicaciones a la medida. El camino ya fue recorrido una vez, así que el trabajo deberá ser mucho más sencillo, pero por supuesto requerirá del trabajo y entusiasmo de todos sus colaboradores para que el compromiso constante conduzca al éxito esperado.

Pero para poder notar resultados reales, no bastará con los esfuerzos internos para lograr productos con calidad e incertidumbre. Es importante que la empresa se mantenga informada acerca de la situación de los competidores internacionales más cercanos y de los impactos que puede llegar a sufrir la industria mexicana dados los conflictos bélicos suscitados en la actualidad entre Estados Unidos (nación que define en gran medida la situación económica de México) y regiones del medio oriente. Las decisiones que se tomen y el tipo de administración que se adopte en la empresa deberán hacer lo posible por conducirla al grupo de empresas que logren continuar creciendo y desarrollándose dentro de los escenarios planteados en las conclusiones del último capítulo de esta investigación.

Glosario

A

Acelerador.- Tecla directa a un botón o a otro elemento de un sistema de información. También denominado "hot key".

Análisis FODA.- Proceso mediante el cual se analizan las fortalezas y debilidades internas de un programa u organización, así como las oportunidades y amenazas que existen fuera del programa.

ASCII.- Uno de los primeros y más usados códigos de caracteres. Existe en versiones de 7 u 8 bits.

Algoritmo.- Conjunto de acciones correctamente definidas para la solución de un problema en un número finito de pasos.

Árbol de decisiones.- Serie de preguntas que se utiliza como herramienta para analizar varias opciones de solución sobre un problema específico.

Asociación.- Es una relación que describe un conjunto de vínculos entre clases. Pueden ser binarias o n-arias, según se implican a dos clases o más. Existe una forma especial de asociación, la agregación, que especifica una relación entre las clases donde el llamado "agregado" indica el todo y el "componente" es una parte del mismo.

Atributo.- Propiedad de un objeto; puede ser valuado o multivaluado.

B

Base de datos.- Conjunto de datos no redundantes, homogéneos, ordenados de una forma determinada, almacenados en soporte magnético u óptico y accesibles por computadora.

Bug.- Error de programación que genera problemas o comportamientos no esperados durante la ejecución de un sistema computacional.

C

Código.- El código de un programa es el conjunto de instrucciones que le hacen funcionar. El "código fuente" es el que escriben los programadores, que luego debe compilarse para producir un código directamente ejecutable por el ordenador.

Composición.- Es un tipo de agregación donde la relación de posesión es tan fuerte como para marcar otro tipo de relación. Las clases en UML tienen un tiempo de vida determinado, en las relaciones de composición, el tiempo de vida de la clase que es parte del todo (o agregado) viene determinado por el

tiempo de vida de la clase que representa el todo, por tanto es equivalente a un atributo, aunque no lo es porque es una clase y puede funcionar como tal en otros casos.

Customización.- Es la creación o adaptación de algo con arreglo a las necesidades del cliente (customer). Una variante, que no significa exactamente lo mismo, pero se puede utilizar en la mayoría de los casos, es "personalización".

D

Dependencia.- Una relación de dependencia se establece entre clases (u objetos) cuando un cambio en el elemento independiente del modelo puede requerir un cambio en el elemento dependiente.

E

Efectividad.- Grado en que un programa ha realizado los cambios deseados o logrado sus objetivos mediante el suministro de servicios.

Eficiencia.- Grado en que un programa ha utilizado recursos apropiadamente y ha completado las actividades de manera oportuna

Evaluación.- Proceso mediante el cual se reúne y analiza información para determinar si un programa lleva a cabo las actividades que planteó y el contexto en que está logrando sus objetivos (por medio de estas actividades). La evaluación es una herramienta que se utiliza para conocer si el programa es lo más efectivo posible o si deben realizarse modificaciones.

Evaluación Bootstrap.- Metodología que mediante prácticas, herramientas y estándares de calidad internacional mide, evalúa y propone mejoras al proceso de desarrollo de software que siguen las Unidades de Producción de Software (UPS) de las empresas.

G

Generalización.- Cuando se establece una relación de este tipo entre dos clases, una es una Superclase y la otra es una Subclase. La subclase comparte la estructura y el comportamiento de la superclase. Puede haber más de una clase que se comporte como subclase.

H

Hardware.- La parte física de la computadora (placa, microprocesador, tarjetas, monitor).

I

ISO.- Organización internacional para la estandarización. La red de institutos nacionales de estandarización de 140 países asociados con organizaciones internacionales representantes de gobiernos, industrias, negocios y consumidores.

Ingeniería de software.- Disciplina que integra métodos, herramientas y procedimientos mediante los cuales se puede desarrollar software eficiente y de calidad en los costos y tiempos estimados.

Interfaz.- Elemento compartido entre dos partes para que interactúen o se comuniquen entre sí. Se pueden considerar como las reglas existentes para establecer dicha comunicación.

Interfaz(2).- Representa el uso de un tipo para describir el comportamiento visible externamente de cualquier elemento del modelo.

L

Lenguaje de programación.- Sistema de escritura para la descripción precisa de algoritmos o programas informáticos.

Lluvia de ideas.- Consiste en una actividad de grupo que permite a los participantes generar ideas, hacer preguntas, proponer soluciones y llegar a acuerdos comunes.

M

Metas .- Son objetivos muy específicos a corto plazo y que pueden ser medidos en términos numéricos. Deben ser medibles, apropiadas, temporales, específicas y realistas y pertenecen a un componente específico del programa.

Método PERT.- Proceso administrativo de planeación, programación, ejecución y control de todas y cada una de las actividades componentes de un proyecto que debe desarrollarse dentro de un tiempo crítico y al costo óptimo.

Módulo.- Unidad de programa que puede ser compilada y unida a otros módulos para formar un programa completo. También lo podemos definir como una parte separable de un programa.

O

Objetivos.- Resultados finales que se esperan lograr por una organización o un individuo. Los objetivos determinan el tipo y enfoque de las actividades que una organización o miembro del personal efectuará con el propósito de alcanzar los resultados deseados.

Objeto.- Unidad que contiene datos y las funciones que operan sobre esos datos; conjunto complejo de datos y programas que poseen estructura y forman parte de una organización.

P

Performance.- Rendimiento, calidad de funcionamiento, desempeño, operación; cumplimiento, rendimiento.

Planeación del trabajo (planeación operativa).- Proceso mediante el cual la organización decide qué actividades se llevarán a cabo, qué departamento o persona las realizará, los recursos que requerirán y el tiempo en el cual se realizarán. La planeación del trabajo generalmente cubre un período de un año o menos.

Programa.- Conjunto de sentencias escritas en un determinado lenguaje para satisfacer un algoritmo en una computadora. Un programa puede ser ejecutado de forma directa. En ciertos contextos programa y software serían equivalentes.

Programación estructurada.- Técnica o estilo de programación que tiene como principios básicos las estructuras de control y la modularidad.

Programación orientada a objetos (POO).- Técnica o estilo de programación que utiliza objetos como bloque esencial de construcción.

R

Recursos.- Medios disponibles para efectuar las actividades planeadas, tales como personal, equipos y dinero.

Red local.- (LAN – Local Area Net), red de área local. Una red de computadoras, dispersas por un edificio o incluso por toda una ciudad.

Relación de Refinamiento.- Es una relación entre dos elementos donde uno de ellos especifica de forma completa al otro que ya ha sido especificado con cierto detalle.

Ruta crítica.- Es una cadena de actividades críticas, es una ruta que identifica todas las actividades críticas del proyecto.

S

Servidor.- Se da este nombre a un ordenador que realiza funciones de asistencia a algún proceso cuya orden proviene de otra máquina (llamada cliente).

Software.- Son los programas de computadoras, que permiten realizar tareas al

T

Tareas.- Actividades divididas en acciones o responsabilidades específicas.

Tarea crítica.- Es una tarea tal, que si hay una demora en su comienzo entonces causará una demora en la terminación del proyecto.

Tipo.- Es un descriptor de objetos que tiene un estado abstracto y especificaciones de operaciones pero no su implementación. Un tipo establece una especificación de comportamiento para las clases.

U

Usuario.- Persona que utiliza la computadora para realizar un trabajo y obtener de ella resultados.

TESIS CON
FALLA DE ORIGEN

Bibliografía

- ☒ ACKOFF, Russel L., "El arte de resolver problemas", Ed. Limusa, México, 1981
- ☒ ACKOFF, Russel L., "Planificación de la empresa del futuro" Ed. Limusa, México, 1981
- ☒ BOOCH, GRADY, "Análisis y Diseño Orientado a Objetos", Ed. Addison-Wesley / Díaz de Santos, 2da edición, México, 1996
- ☒ BOOCH, JACOBSON, RUMBAUGH, "Developing Applications with Visual Basic and UML", Ed. Addison Wesley, EUA, 2000
- ☒ CARNEGIE MELLON UNIVERSITY. Software Engineering Institute, "The Capability Maturity Model. Guidelines for Improving the Software Process", Ed. Addison Wesley, USA, 1997
- ☒ WEST Churchman, C., "El enfoque de sistemas para la toma de decisiones", Ed. Diana, 18ª impresión, México 1995
- ☒ JALOTE, Pankaj "CMM in Practice", Ed. Addison Wesley, EUA, 1999
- ☒ JALOTE, Pankaj, "Software Project Management in Practice", Ed. Addison Wesley, EUA, 2002
- ☒ ROBBINS Stephen P., "Administración, teoría y práctica" Ed. Prentice Hall Hispanoamericana, México, 1987
- ☒ RUMBAUGH, JACOBSON, BOOCH, "The Unified Modeling Language Reference Manual", Ed. Addison Wesley, EUA, 1998
- ☒ SALDIVAR, Antonio, "Planeación financiera de la empresa" Ed. Trillas, México, 1986
- ☒ SENN James A., "Análisis y diseño de sistemas de información" Ed. Mc Graw Hill, México 1990
- ☒ VAN DER HEIJDEN, Kees, "Escenarios", Ed. Panorama, México D.F., 1998

Otras fuentes de consulta

- ☐ certum, 1995, "Control y costeo de proyectos"
- ☐ certum, 1994, "Guías para la planeación y control de proyectos"
- ☐ certum, 1998, "Proceso de asignación y control de proyectos"
- ☐ Memorias del primer seminario internacional de calidad en desarrollo de software 2002
- ☐ <http://www.certum.com>
- ☐ <http://www.rational.com/uml>
- ☐ http://www.objectsbydesign.com/tools/modeling_tools_sp.html
- ☐ <http://www.aceproject.org/main/espanol/em/emd01.htm>
- ☐ <http://gidis.ing.unlpam.edu.ar/personas/glafuente/uml>

TESIS CON
FALLA DE ORIGEN