

INSTITUTO TECNOLÓGICO Y DE ESTUDIOS
SUPERIORES DE MONTERREY

CAMPUS MONTERREY

PROGRAMA DE GRADUADOS DE LA DIVISION DE
ELECTRONICA, COMPUTACION, INFORMACION Y
COMUNICACIONES



ITESM

Campus Monterrey

INDICADORES DE FALLA EN EL DESARROLLO
DE UN NUEVO SISTEMA DE INFORMACION

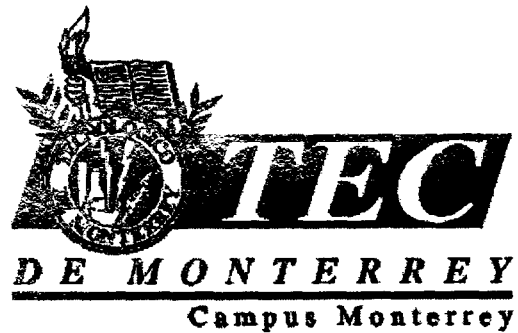
T E S I S

MAESTRIA EN ADMINISTRACION
DE TECNOLOGIAS DE INFORMACION

POR
EDUARDO SAMANIEGO HERNANDEZ

NOVIEMBRE DE 2000

INDICADORES DE FALLA EN EL DESARROLLO
DE UN NUEVO SISTEMA DE INFORMACION



TESIS

MAESTRIA EN ADMINISTRACION DE TECNOLOGIAS DE INFORMACION

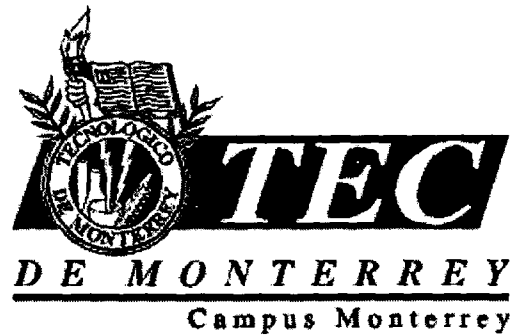
INSTITUTO TECNOLOGICO Y DE ESTUDIOS SUPERIORES DE
MONTERREY

POR

EDUARDO SAMANIEGO HERNANDEZ

NOVIEMBRE DE 2000

INDICADORES DE FALLA EN EL DESARROLLO
DE UN NUEVO SISTEMA DE INFORMACION



POR

EDUARDO SAMANIEGO HERNANDEZ

TESIS

Presentada a la División de Electrónica,
Computación, Información y Comunicaciones
Este trabajo es requisito parcial para obtener el Título de
Maestro en Administración de Tecnologías de Información

INSTITUTO TECNOLÓGICO Y DE ESTUDIOS SUPERIORES DE
MONTERREY

NOVIEMBRE DE 2000

DEDICATORIA

A mis padres Alicia y Eduardo

A mis hermanos Daniela, Bertha Alicia y Andrés

A mi esposa Silvia

RECONOCIMIENTOS

Mi más sincero agradecimiento a mi asesor Dr. David A. Alanís Dávila por compartir conmigo su experiencia, tiempo y conocimientos; a mis sinodales Ing.Martín de Jesús González Martínez y el Ing.Gustavo Cervantes Ornelas por sus sugerencias y aportaciones para enriquecer esta tesis.

Agradezco especialmente a la Lic.Dolores Guadalupe Lankenau Caballero e Ing.Martín de Jesús González Martínez por haberme dado la oportunidad de trabajar con ellos y poder realizar este sueño.

RESUMEN

INDICADORES DE FALLA EN EL DESARROLLO DE UN NUEVO SISTEMA DE INFORMACION

Esta tesis busca encontrar aquellos puntos que pueden en un momento determinado, afectar el desarrollo de un sistema de información. Como cualquier proyecto de cualquier índole, un proyecto de desarrollo de un sistema de información no está exento de caer en fallas, las cuáles pueden modificar el presupuesto, el tiempo de desarrollo, la tecnología utilizada, el personal del equipo de trabajo, etc., pudiendo ocasionar que se retrase su culminación o en el peor de los casos, que el proyecto llegue a cancelarse.

Uno de los hábitos más comúnmente encontrados dentro de los equipos de trabajo, es que dentro del equipo no existe una adecuada comunicación y relación entre sus integrantes, formando con esto una atmósfera de trabajo no muy agradable. Pero no solamente cuestiones relacionadas con el equipo de trabajo son las que pueden perjudicar al proyecto que desarrollamos, podemos encontrar problemas tanto internos al grupo de trabajo, al departamento o a la empresa, así como también problemas externos que pueden estar fuera de nuestro control.

En esta tesis, al identificar los indicadores más comunes que afectan al desarrollo de un sistema de información, buscaremos aquellas áreas que puedan ser clave y sujetas de mejora de acuerdo a las opiniones de las personas que nos otorgaron parte de su tiempo para poder contestar la encuesta que les hicimos llegar. En este trabajo abarcamos los siguientes puntos:

- Ciclo de vida de desarrollo de sistemas
- Administración de proyectos
- Equipo de trabajo

Dentro de estos puntos principales, nos enfocamos a los criterios de asignación de proyectos al personal, comunicación dentro y fuera del equipo de trabajo, retroalimentación por parte del líder del proyecto, modelos de desarrollo de sistemas de información, responsabilidades de cada uno de los miembros de un equipo, y por supuesto, aquellos indicadores que se presentan y afectan nuestro trabajo.

TABLA DE CONTENIDO

	Página
Lista de Tablas	VII
Lista de Figuras	VIII
Capítulo 1 Introducción	
1.1 Definición del problema	1
1.2 ¿Por qué desarrollar un proyecto de informática?	3
1.3 Objetivo y restricciones de la tesis	3
1.4 Aportación de la investigación	4
1.5 Conclusiones	4
Capítulo 2 Ciclo de vida de un proyecto de desarrollo de sistemas	
2.1 Introducción	6
2.2 Definición del ciclo de vida de desarrollo de sistemas	6
2.3 Modelos a seguir para desarrollar el sistema	8
2.3.1 Clásico	10
2.3.2 Cascada Pura	12
2.3.3 Codificar y Corregir (Code and Fix)	13
2.3.4 Espiral	14
2.3.5 Cascadas Modificadas (Sashimi)	16
2.3.6 Cascadas Modificadas (Subproyectos)	17
2.3.7 Prototipo Evolutivo	17
2.3.8 Entrega por Etapas	18
2.4 Conclusiones	19
Capítulo 3 Administración de Proyectos	
3.1 Introducción	21
3.2 ¿Qué es la Administración de Proyectos?	21
3.3 Características de los Proyectos de Informática	23
3.4 Antes de comenzar el Proyecto	24
3.5 ¿Por qué fracasan los Proyectos de Informática?	25
3.6 Indicadores de falla en los Proyectos de Informática	26
3.7 Conclusiones	32
Capítulo 4 Equipo de trabajo	
4.1 Introducción	34
4.2 El Cliente	34
4.3 El Líder del Proyecto	37
4.4 El Equipo de Desarrollo	40
4.5 Relación y responsabilidades de los participantes	44
4.6 Conclusiones	48

	Página
Capítulo 5 Investigación de Campo	
5.1 Introducción	49
5.2 Objetivos de la Investigación	49
5.3 Criterios de selección de empresas	50
5.4 Muestra utilizada	50
5.5 Herramientas y variables de la investigación	51
5.6 Conclusiones	52
Capítulo 6 Resultados de la Investigación de campo	
6.1 Introducción	53
6.2 Criterio de asignación de nuevos proyectos	53
6.3 Desarrollo de sistemas en el último año	54
6.4 Participación directa del usuario	60
6.5 Inicio del proyecto	65
6.6 Comunicación y retroalimentación	68
6.7 Metodologías de desarrollo	76
6.8 Indicadores de falla presentados en el último año en el desarrollo de sistemas de información	77
6.9 Origen de las fallas	87
6.10 Areas de mejora	88
6.11 Conclusiones	89
Capítulo 7 Conclusiones y Trabajos Futuros	90
Anexo 1 Instrumento de la Investigación	94
Anexo 2 Variables de la Investigación	101
Anexo 3 Test de supervivencia	104
Anexo 4 Tabla comparativa de Modelos de Desarrollo	107
Anexo 5 Problemas en el ambiente del proyecto	109
Anexo 6 Elementos de un proyecto exitoso	110
Anexo 7 Mapa de errores y aciertos	111
Anexo 8 Tabla de situaciones	112
Anexo 9 Arbol de decisiones	118
Bibliografía	119

	Página
Vita	122

Lista de Tablas		Página
Tabla 3.1	Sumario de errores clásicos	31
Tabla 4.1	Directrices básicas para miembros y responsables de equipos	47

Lista de Figuras

		Página
Fig.2.1	Estado de las etapas en el ciclo de vida de desarrollo de sistemas	8
Fig.2.2	Modelo general de un SDLC	9
Fig.2.3	Ciclo de vida clásico de desarrollo de sistemas	11
Fig.2.4	Modelo de cascada pura	12
Fig.2.5	Modelo de codificar y corregir (Code and Fix)	13
Fig.2.6	Modelo en espiral	14
Fig.2.7	Modelo en espiral simple	15
Fig.2.8	Modelo de cascadas modificadas	16
Fig.2.9	Modelo de subproyectos	17
Fig.2.10	Modelo de prototipo evolutivo	18
Fig.2.11	Modelo de entrega por etapas	19
Fig.3.1	Origen de errores	26
Fig.4.1	Puntos de vista de un fenómeno	35
Fig.4.2	Métodos típicos de recopilación de requerimientos y orientados al cliente	36
Fig.4.3	Organización de una administración de proyectos	38
Fig.4.4	Trabajo en equipo	41
Fig.4.5	Desarrollador de software motivado	42
Fig.6.1	Asignación de nuevos proyectos	53
Fig.6.2.1	Entrega de sistemas en el último año	54
Fig.6.2.2	Modificaciones a los requerimientos	54
Fig.6.2.3	Cumplimiento de expectativas del usuario	55
Fig.6.2.4	Uso adecuado de los sistemas	55
Fig.6.2.5	Modificación por al menos 1 problema presentado	56
Fig.6.2.6	Modificaciones por 2 a 5 problemas presentados	56
Fig.6.2.7	Modificaciones por 6 a 10 problemas	57
Fig.6.2.8	Problemas ocasionados por la Tecnología de Información	57
Fig.6.2.9	Modificaciones por falta de habilidades del equipo de trabajo	58
Fig.6.2.10	Modificaciones presentadas por problemas económicos	58
Fig.6.2.11	Modificaciones presentadas por falta de personal	59
Fig.6.3.1	Participación del usuario en la Identificación de las necesidades	60
Fig.6.3.2	Participación del usuario en el estudio de factibilidad	60
Fig.6.3.3	Participación del usuario en el análisis	61
Fig.6.3.4	Participación del usuario en el diseño lógico	61
Fig.6.3.5	Participación del usuario en el diseño físico	62
Fig.6.3.6	Participación del usuario en la codificación	62
Fig.6.3.7	Participación del usuario en la etapa de pruebas	63
Fig.6.3.8	Participación del usuario en la etapa de instalación	63

	Página
Fig.6.3.9	Participación del usuario en la etapa de mantenimiento 64
Fig.6.4.1	Uso de metodologías 65
Fig.6.4.2	Definición de actividades 65
Fig.6.4.3	Selección y evaluación de la metodología 66
Fig.6.4.4	Conocimiento de las fases de la metodología 66
Fig.6.4.5	Herramientas necesarias 67
Fig.6.5.1	Inicio de requerimientos 68
Fig.6.5.2	Misión del departamento de informática 68
Fig.6.5.3	Retroalimentación al equipo de trabajo 69
Fig.6.5.4	Decisiones de la alta gerencia 69
Fig.6.5.5	Canales bien definidos 70
Fig.6.5.6	Disponibilidad 70
Fig.6.5.7	Revisión antes de liberación 71
Fig.6.5.8	Compatibilidad 71
Fig.6.5.9	Participación al término del trabajo 72
Fig.6.5.10	Resolución por sí mismo 72
Fig.6.5.11	Administración por parte del líder del proyecto 73
Fig.6.5.12	Metodología a seguir definida por el líder del proyecto 73
Fig.6.5.13	Definición de plataforma 74
Fig.6.5.14	Autorización del presupuesto 74
Fig.6.5.15	Usuario al término del sistema 75
Fig.6.6.1	Metodología favorita 76
Fig.6.6.2	Metodología favorita seleccionada 76
Fig.6.7.1	No uso de metodología 77
Fig.6.7.2	Cambios en los requerimientos 77
Fig.6.7.3	Fechas de inicio y/o entrega 78
Fig.6.7.4	Necesidades del usuario 78
Fig.6.7.5	No deseo del sistema 79
Fig.6.7.6	Percepciones de las necesidades 79
Fig.6.7.7	Presiones políticas 80
Fig.6.7.8	Enfoque tecnológico débil 80
Fig.6.7.9	Habilidades del equipo de trabajo 81
Fig.6.7.10	Carencia de métricas de desempeño 81
Fig.6.7.11	Carencia de Apoyo directivo 82
Fig.6.7.12	Ruido en el área de desarrollo 82
Fig.6.7.13	Eliminación de pruebas de calidad 83
Fig.6.7.14	No existe relación con la meta y visión de la empresa 83
Fig.6.7.15	Responsable directo del proyecto 84
Fig.6.7.16	Estructura de desarrollo débil 84
Fig.6.7.17	Duplicidad de funciones 85
Fig.6.7.18	Análisis de riesgo deficiente 85
Fig.6.7.19	Revisión de proyectos anteriores 86
Fig.6.8	Origen de las fallas 87
Fig.6.9	Areas de mejora 88

		Página
Fig.7.1	Problemas en el ambiente del proyecto	93
Fig.7.2	Elementos de un proyecto exitoso	94
Fig.7.3	Mapa conceptual de errores y aciertos	95

Capítulo I Introducción

Hoy en la actualidad en el ambiente tanto laboral como en las universidades, escuchamos la palabra "proyecto". Cuando estamos trabajando en una organización por lo general se nos asigna a trabajar dentro de un proyecto, ya sea que ya esté en marcha o se vaya a comenzar a trabajar en él. Así mismo en nuestra preparación académica a lo largo de nuestra estancia en la universidad, debemos de hacer proyectos para ser evaluados en las materias que nos imparten como parte de nuestra preparación profesional. Pero a todo esto, ¿qué es un proyecto?.

Un proyecto de acuerdo a Cervantes (1996), citado por Valdéz (1997), "Es un trabajo que se ejecuta por una sola vez, que tiene un inicio y un final, un objetivo especificado con claridad, un presupuesto establecido y una organización quizá temporal".

De la definición de proyecto que acabamos de citar, podemos darnos cuenta que un proyecto tiene una duración de vida en cuanto a tiempo se refiere. Ahora bien, para llevar a cabo un proyecto, se debe de reunir ciertas características, que de acuerdo con López (1998), deben de contemplar tanto el conocimiento de los individuos participantes, como sus habilidades técnicas necesarias para llevar a cabo las actividades de un proyecto.

Cuando un proyecto está propuesto para llevarse a cabo, la necesidad de que éste se realice obedece exclusivamente a satisfacer las necesidades del cliente, ese es el objetivo principal del nacimiento de un proyecto, que ayude al cliente (y al hablar de cliente no necesariamente estamos refiriéndonos a alguien externo a la compañía, un cliente puede ser tanto interno como externo a la organización) a satisfacer esas necesidades que él tiene.

1.1 Definición del Problema

Una de las mayores preocupaciones de toda empresa, es el derroche de dinero en inversiones que finalmente no producen lo que se espera, o bien simplemente se quedan en el camino.

El desarrollo de aplicaciones de sistemas de información, la adquisición de nueva tecnología, la capacitación en informática, la compra de licencias de software, son sólo algunos de los puntos en los que una empresa decide poner su dinero, con el único fin de poder trabajar en forma correcta y eficaz para poder ofrecer a sus clientes productos y servicios con la mejor calidad posible.

En toda etapa de desarrollo de un proyecto de informática, no solamente debemos de concentrarnos en tener a nuestro alcance los elementos necesarios (recursos financieros, tecnológicos, humanos, legales, etc.), sino que también debemos tomar en cuenta que existen riesgos en todo proyecto y de acuerdo con King (1992), cualquier tipo de proyecto, fracasará en su intento si no se conoce exactamente los requerimientos del cliente, o bien en su defecto, éste llega a implementarse en un tiempo posterior al planeado, y ocasione que no sea efectivo, o simplemente, sobrepase el presupuesto estimado.

Una vez que ya se tiene definido que es lo que se quiere que el proyecto haga para el cliente, y una vez que se sabe que es lo que se debe de hacer, se debe de formar un equipo de trabajo para poder llevarlo a cabo, y un equipo de trabajo necesita de ser guiado y administrado para poder cumplir con sus tareas.

¿Pero qué ocurre cuando se cae en algún error por muy simple que sea?

La única parte perjudicada en todo esto es la misma empresa, ya que se retrasa en su planeación de desarrollo, y además puede caer en el gasto innecesario o simplemente gastar más de lo presupuestado.

He aquí el por qué la importancia de estudiar los puntos que ocasionan que se den los problemas que desvían la intención inicial del nacimiento de un proyecto, simplemente con el propósito de que nos demos una idea de todo aquello que nos puede afectar y tomarlo en cuenta en algún proyecto futuro.

La idea inicial es estudiar tanto proyectos que ya se llevaron a cabo y analizar cuales fueron sus problemas, así como proyectos que se lleven a cabo actualmente, y ver de alguna manera cómo se han resuelto los problemas en los proyectos terminados.

Es muy difícil que un proyecto de informática, no sufra al menos un tropezón, por muy pequeño que sea, pero para darnos una idea más profunda sobre que es lo que está detrás de un sistema de información, detrás de la utilización de alguna tecnología, veremos que es todo aquello que hace posible que los utilicemos, pero sobre todo, proporcionar un panorama que ayude a tratar de prevenir que se presente algún tropiezo, claro está, como todo en la vida, siempre estaremos expuestos a caer, pero la idea es levantarse lo más pronto posible y al menor costo.

El llevar a cabo un análisis de los problemas más comunes que se presentan en la vida de un proyecto de informática en las empresas, así como la propuesta de varias alternativas de solución para poder sacar adelante un proyecto cuando este llegue a presentar síntomas de caer en algún error, nos ayudará a tener alguna referencia de alguna situación anterior para poder tener un marco de referencia sobre como poder tomar un camino para solucionar el conflicto.

1.2 ¿Por qué desarrollar un proyecto de informática?

Si una empresa quiere lograr una ventaja competitiva, debe tener unas estrategias de desempeño bien definidas con el objetivo de poder satisfacer de la manera más eficiente las necesidades de sus clientes, además de involucrarse con la tecnología de información, ya que ésta le proporciona a la empresa una herramienta poderosa para proporcionar un mejor servicio y/o producto a sus clientes, además de tener la posibilidad de darles un valor agregado, para que éste este satisfecho en la mayor medida posible (Flores, 1993).

Cualquier proyecto de informática (sea desarrollo de nuevas aplicaciones, actualización de software, actualización de equipo, instalación de redes, etc.) tienen como objetivo principal que la empresa maneje información en su proceso de toma de decisiones, para que sus directivos marquen el rumbo a seguir, sabiendo que tienen la tecnología y sistemas de información adecuados para lograrlo de la mejor manera y con los menores costos posibles, siempre buscando la satisfacción del cliente (Gómez, 1992).

De lo anteriormente dicho podemos percibir que para que una empresa decida llevar a cabo proyectos de informática para aplicarlos en alguna de sus actividades o procesos ya sea productivo o de administración, debe de llevarse a cabo una serie de análisis para saber si es factible que se desarrolle o no, con el propósito de intentar evitar caer en errores que ocasionen que se eroguen gastos innecesarios, o bien, simplemente llegue a desviarse el desarrollo del mismo por cualquier situación no contemplada.

1.3 Objetivo y restricciones de la Tesis

El objetivo de la Tesis es analizar diferentes situaciones que se presentan en el desarrollo de un proyecto de informática, enfocándose en el área de desarrollo de sistemas de información, con el propósito de proponer algunas áreas de mejora para futuras referencias y que éstas sirvan como base para la solución de los conflictos.

Es conveniente mencionar que la Tesis tendrá las siguientes restricciones:

- La disponibilidad de las personas que se entrevisten para la recopilación de información, experiencias y análisis de casos
- La investigación se llevará a cabo en la ciudad de Monterrey, N.L. y su área metropolitana, en empresas cuyo giro principal sea el Análisis de Sistemas y Procesamiento Informático tanto empresas medianas y pequeñas, con el fin

de poder observar posibles diferencias en sus planeaciones, formas de trabajo, etc.

- La confidencialidad que pueda tener cierta empresa con respecto a la información que se solicite referente a algún proyecto de informática realizado anteriormente o bien que se esté llevando a cabo
- La aplicación de las recomendaciones así como el posible efecto en su implementación esta fuera del alcance de esta tesis.

De acuerdo a las respuestas de cada una de las personas que contesten las encuestas, se formularán las conclusiones finales de la investigación, sabiendo de antemano, que pudiera presentarse la situación que algunas de estas personas no contesten todas las preguntas contenidas en la encuesta, dichas respuestas en blanco no serán tomadas en cuenta para los resultados finales.

1.4 Aportación de la investigación

Al terminar la investigación de esta tesis, se podrá contar con un panorama de aquellos contratiempos que ocasionan que un proyecto de informática tenga un retraso o bien lleguen a ser cancelados. Así mismo, al saber cuales son aquellos puntos que surgen y afectan en una manera negativa en el desarrollo de un proyecto, se mostraran diferentes áreas de mejora para poder resolver esos conflictos, pero más que resolver, tratar de evitar caer en esos errores.

La contribución que se espera se tenga al final del trabajo, es proporcionarle básicamente a los líderes de proyecto de informática, una herramienta que le pueda ayudar a tomar en cuenta los riesgos en los que puede caer en el transcurso de la etapa de desarrollo del mismo, así como darle una idea de qué acciones puede llevar a cabo para que esto no afecte en gran medida a la organización, sobre todo porque toda desviación del mismo, lleva finalmente a la empresa a tener que realizar una inversión mayor a la que tenía presupuestada.

1.5 Conclusiones

Posiblemente en el desarrollo de esta tesis, y de acuerdo a las respuestas en las encuestas, podríamos encontrar, que en algunas empresas, no necesariamente en todas, no utilicen alguna de las metodologías de desarrollo más conocidas y sean ellos quienes hayan definido alguna metodología especial para trabajar. También podríamos ver que se pueden presentar problemas diferentes a los que se van a tratar en los capítulos siguientes.

Debido a que no se tiene una obligación de contestar las preguntas de la encuesta, el análisis de dichas encuestas quedará sujeto a lo que el personal seleccionado pueda contestar, y en el tiempo en que puedan darnos la retroalimentación de la misma, tomando como verdad todo lo que nos digan, sin poder tener una herramienta que nos permita comprobar dichas respuestas.

Capítulo II

Ciclo de vida de un proyecto de desarrollo de sistemas

2.1 Introducción

Antes de comenzar a desarrollar cualquier trabajo que se nos asigne, lo primero que hacemos es analizar que es lo que queremos hacer, determinar que necesitamos para poder cumplir con la meta, y sobre todo saber cómo lo vamos a conseguir.

Si no seleccionamos adecuadamente la metodología que vamos a seguir para poder desarrollar el trabajo, tendremos contratiempos tarde o temprano para poder cumplir con lo que se nos encomendó. Aquí reviste la importancia de los modelos de ciclo de vida de desarrollo de sistemas, ya que cada uno de estos nos marca diferentes etapas que se deben de desarrollar para poder tener un sistema de información tal y como lo quiere el cliente.

2.2 Definición de Ciclo de Vida de Desarrollo de Sistemas

De acuerdo a lo mencionado por Dorfman y Thayer (1997), los sistemas de software son desarrollados a través de una estructura organizacional llamada Proyecto de Software. Un Proyecto de Software exitoso se deriva de la planeación de un producto a través de un esquema y un presupuesto, conociendo las funciones y calidad requeridas. Dentro de este desarrollo, se encuentran entre algunas, las siguientes actividades:

- Análisis de requerimientos de software
- Diseño de software
- Métodos de programación
- Procedimientos de prueba
- Verificación y validación
- Configuración del software
- Aseguramiento de la calidad del software

Como se puede ver son actividades diversas, y cada una tiene una meta final, la cuál sirve como entrada a una etapa subsecuente. Es indispensable que cada una de estas etapas se lleven a cabo para que el proyecto vaya por buen camino, y así poder obtener un buen producto.

En la actualidad, las empresas necesitan integrar a sus operaciones las aplicaciones de software necesarias para alcanzar a tener una buena penetración en la competencia respecto a sus rivales. La información es de vital

importancia en estos tiempos para la toma de decisiones, y esta información se obtiene de muchas formas, una de ellas, la cuál es cada día más penetrante en más y más empresas, es la que proporcionan los sistemas de información. Un sistema de información se desarrolla en varias etapas, y en este capítulo veremos algunos modelos de ciclo de vida de desarrollo de sistemas.

Para comenzar a darnos una idea de lo que es el ciclo de vida de un proyecto de desarrollo de sistemas, veamos lo que dice McConnell (1997) en su libro *Desarrollo y Gestión de Proyectos Informáticos*: "La función principal de un modelo de ciclo de vida es establecer el orden en el que se especifica, se realizan los prototipos, se diseña, implementa, revisa, prueba y se realizan otras actividades en un proyecto". Ahora veamos lo que nos dice Benjamin (1978), "la base del éxito de cualquier proyecto de desarrollo está en entender a fondo su proceso mismo".

Analizando lo anteriormente mencionado, se puede decir que al aplicar un modelo de ciclo de vida (de los cuáles existen varios y mencionaremos algunos un poco más adelante), éste nos podrá ayudar a decidir que actividades son las que necesitamos o necesitaremos realizar al llevar a cabo el desarrollo de un sistema (sea nuevo o actualización).

Efectivamente el modelo puede ayudar a identificar lo que debemos de hacer, pero no indica que el desarrollo tenga un éxito asegurado, ya que existen muchos factores que afectan la vida del mismo, y sobre todo, se puede caer en el error de no seleccionar correctamente el modelo adecuado. Y si se elige erróneamente el modelo, puede pasar lo que menciona McConnell (1997), que sea una fuente constante de retrasar del trabajo, trabajo repetitivo, innecesario y frustrante. Hay una frase que puede ayudar a aclarar esto, Lewis Carroll citado por King (1992) dice: "Comienza en el principio, y llegaras hasta el final, entonces detente".

Esto conlleva a darse cuenta que para poder estar seguros que el desarrollo del proyecto podrá tener un buen inicio y un muy buen final, se debe comenzar bien desde un principio, y ese principio, comienza al seleccionar el modelo de ciclo de vida a desarrollar.

Si no se selecciona correctamente y se intenta desarrollar el sistema, es como navegar sin brújula en el Océano Atlántico (Benjamín, 1978). Esto da a entender, que si no se conoce cuáles son los pasos que debemos de seguir, saber cuál es la secuencia de cada una de las etapas, es muy difícil, pero no imposible, que un proyecto pueda tener un buen desarrollo. Cabe la posibilidad de que sí pueda llegar a desarrollarse para los fines que se desea, pero difícilmente se logrará tener en un tiempo y sobre todo a un costo adecuado, si se elige el desarrollo sin seguir algún modelo de ciclo de vida, seguramente los trabajos serán muy poco profesionales y a tendrá altos costos para la organización.

Esencialmente, todos los SDM (*System Development Methodology*) son un conjunto de reglas para subdividir el desarrollo del sistema dentro de un conjunto de tareas discretas fáciles de manejar (King, 1992). Así al ver esta definición se puede apreciar que al seleccionar y aplicar un modelo de ciclo de vida, éste ayudara a dividir todo el amplio mundo de actividades que envolverán al proyecto, en pequeñas tareas que se irán llevando a cabo paso a paso con el fin único de que se pueda llevar un adecuado manejo del proyecto.

2.3 Modelos a seguir para desarrollar el sistema.

Un ciclo de vida de desarrollo de sistemas (también conocido como SDLC *System Development Life Cycle*), es un perfil de un proceso que ayuda a desarrollar con éxito un sistema de información (Parsons y Oja, 1995). En el ciclo de vida de desarrollo de sistemas, este contiene una serie de estados, en donde cada uno es distinguido por las actividades desarrolladas en él y por un producto final, el cuál es tomado como la entrada para el siguiente estado o etapa del ciclo de vida (King, 1992).

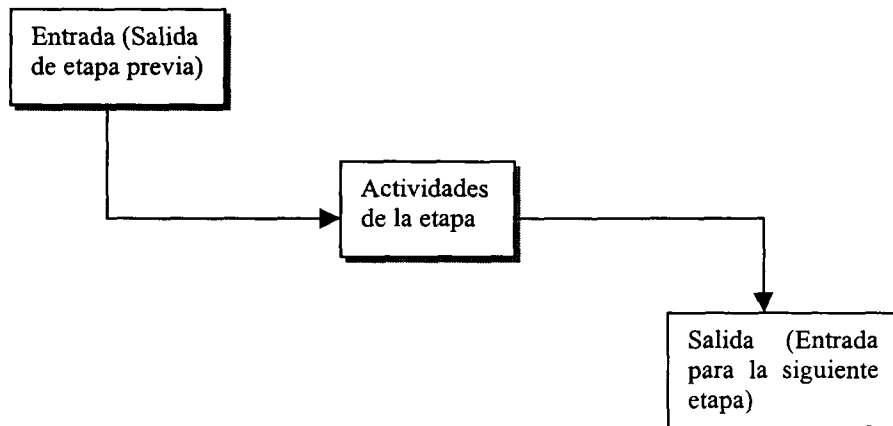


Fig. 2.1 Estados de las etapas en el ciclo de vida de desarrollo de sistemas
[Fuente: King, 1992]

A continuación se muestra un modelo general sobre un SDLC.

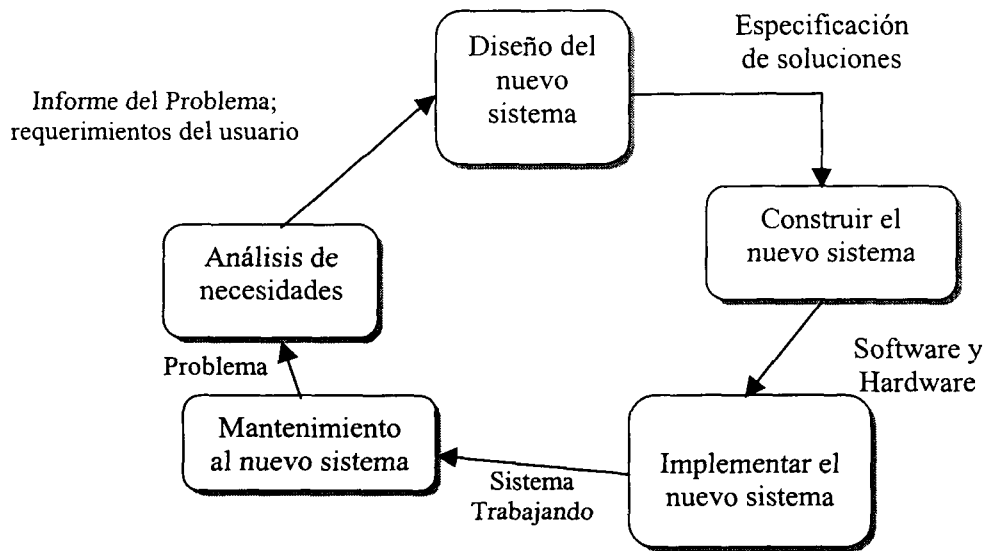


Fig. 2.2 Modelo general de un SDLC
[Fuente: Parsons y Oja, 1995]

Existen varias variaciones en algunos modelos, pero muchos de ellos son muy similares o tienen poca variación en el modelo anterior (Parsons y Oja, 1995). Como se mencionó anteriormente, existen varios modelos de ciclo de vida que pueden seleccionarse y que es responsabilidad del líder del proyecto, analizarlos y escoger el más idóneo de acuerdo a las características del proyecto que se desea desarrollar.

Algunos de los modelos de Ciclo de Vida de Desarrollo de Sistemas son:

- Clásico
- Cascada Pura
- Modificar y Corregir
- Espiral
- Sashimi
- Subproyectos
- Prototipo Evolutivo
- Entrega por etapas

Antes de poder seleccionar el modelo adecuado para el proyecto que se desea llevar a cabo, King (1992) nos menciona que el administrador o líder del proyecto, debe responder a las siguientes preguntas:

- ¿Que? (estamos creando)

- ¿Cuándo? (entregaremos el proyecto)
- ¿Quién? (es el responsable de cada etapa)
- ¿Cuánto? (costo del proyecto)

Agregando a las preguntas anteriores, se pueden agregar las siguientes que menciona Benjamin (1978):

- ¿Dónde nos encontramos ahora en esta etapa del proyecto?
- ¿Qué probabilidades de éxito tenemos?

Es recomendable analizar las preguntas anteriores no sólo en el momento de decidir que modelo se va a seguir, si no que también sería adecuado que se tomaran en cuenta en cada una de las etapas del modelo que sigamos. A continuación se presentan los diferentes modelos de ciclo de vida de desarrollo de sistemas que se mencionaron anteriormente. En primer lugar se muestra el modelo propuesto por Yourdon (1988), y al terminar este, se muestran algunos de los propuestos por McConnell (1997) y una variación del modelo de espiral presentado por King (1992).

2.3.1 Modelo de Ciclo de Vida Clásico

Yourdon (1988) nos presenta el siguiente modelo de ciclo de vida, al cuál le llama el "Ciclo de Vida Clásico (*Classical Project Life Cycle*)" En este modelo, se puede apreciar que antes de que se lleve a cabo el diseño, es indispensable que se realice un análisis no solamente de las necesidades o requerimientos del usuario o cliente, el análisis se debe de realizar tanto a los requerimientos del cliente como al estudio del presupuesto necesario, la infraestructura física necesaria, y llevar a cabo un diseño preliminar con los puntos más importantes del proyecto.

Una vez que se realizan estos pasos previos, y si es que el proyecto es autorizado, se prosigue a diseñar ya en una forma completa el proyecto y se continúan con las demás etapas. No por el simple hecho de que alguien quiere un sistema para realizar algunas funciones, quiere decir que todos los proyectos o ideas que se puedan llegar a tener, deben de terminar en un sistema de información, para eso se realiza el análisis, para determinar si es o no es conveniente llevarlo a cabo. Yourdon (1988) dice que este modelo se caracteriza por:

- Tener una fuerte tendencia a la implementación de técnica *bottom - up*
- Progresión secuencial de una actividad a otra

Tal como lo podemos apreciar en la fig. 2.3, no existe un regreso hacia una actividad anterior en ningún momento.

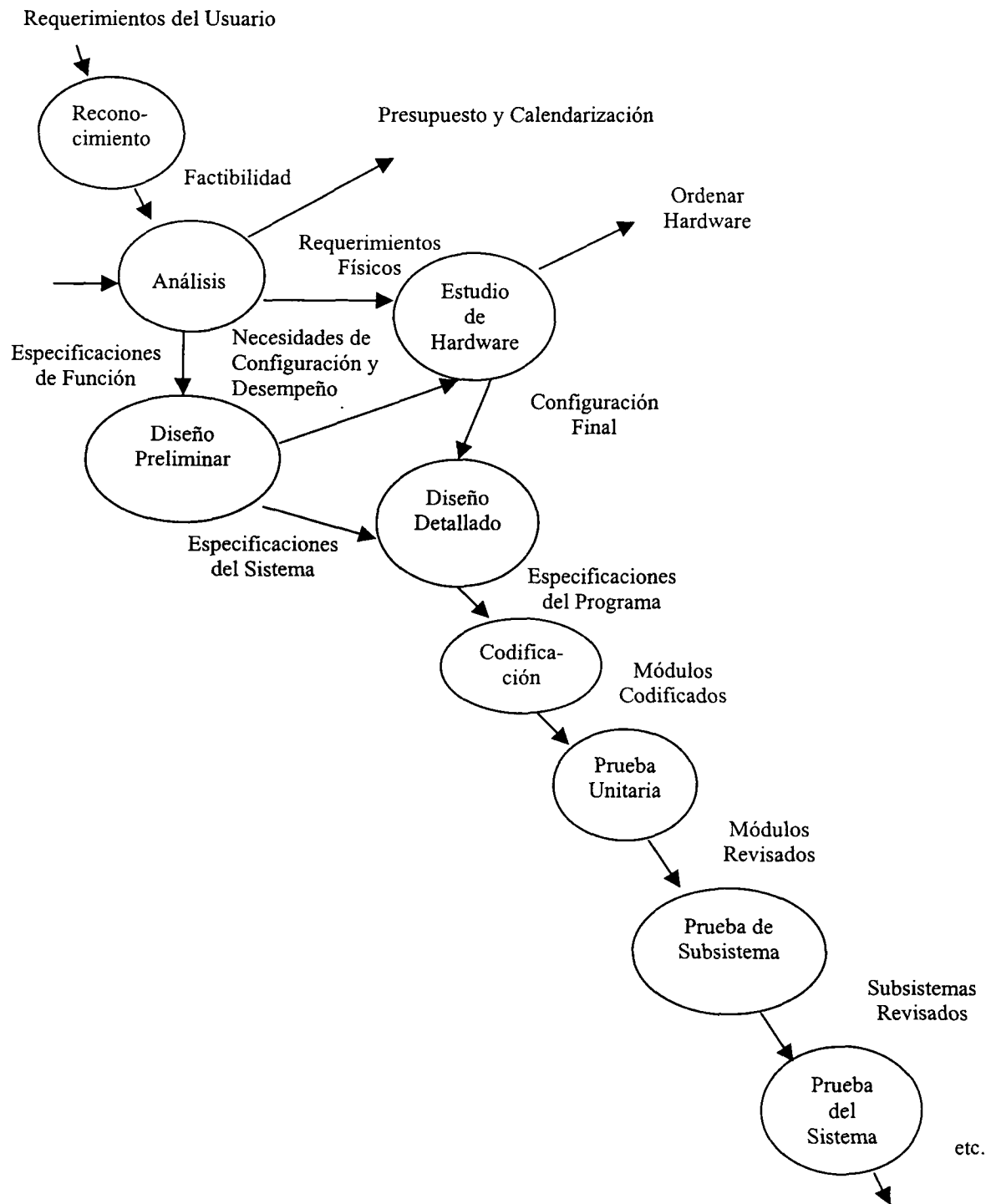


Fig. 2.3 Ciclo de vida clásico de desarrollo de sistemas
[Fuente: Yourdon, 1988]

2.3.2 Modelo de Ciclo de Vida Cascada Pura

Este modelo es muy similar al clásico de Yourdon, para ejemplificarlo veamos la figura 2.4.

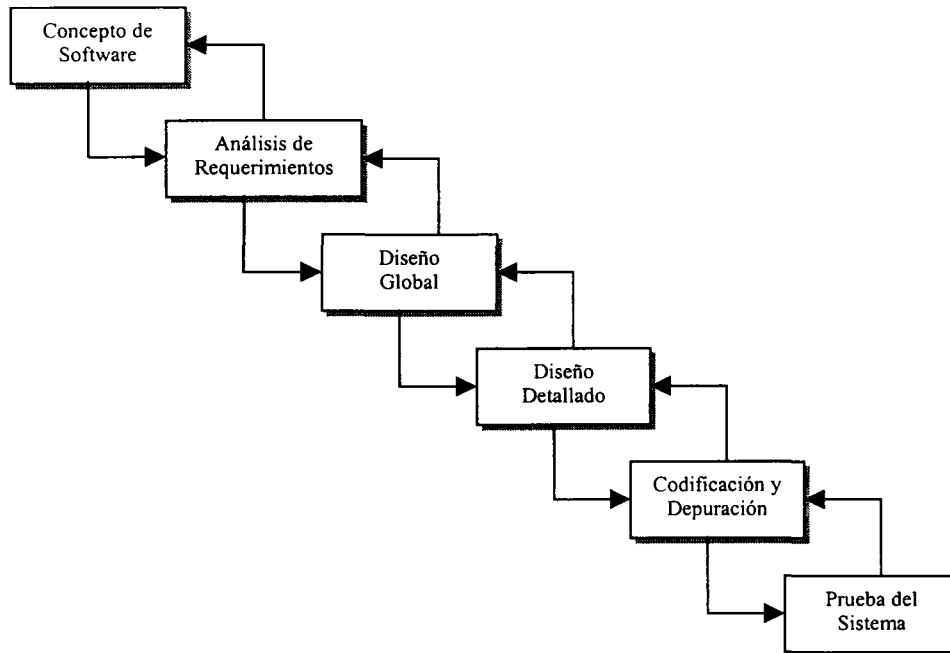


Fig. 2.4 Modelo de cascada pura
[Fuente: McConnell, 1997]

Para McConnell (1997) este es el modelo predecesor a todos los modelos que se puedan encontrar en la actualidad. Es muy parecido al clásico de Yourdon, pero con una diferencia que puede ser a simple vista no muy importante pero más sin embargo si lo es. En este modelo de McConnell, existe la posibilidad de un regreso, no a una etapa anterior, si no a volver a realizarla nuevamente hasta que esté lista y pueda pasarse a desarrollar la siguiente actividad.

En este mismo modelo, Martínez (1996), especifica que en este modelo, el concepto fundamental es el de terminar una etapa con verificación antes de empezar la siguiente, evitando con esto la propagación de fallas, en el supuesto de que una falla encontrada en las etapas iniciales es mas barata en costo y daño que en etapas posteriores.

Como mencionaba anteriormente, cada modelo se puede aplicar a los proyectos de acuerdo a las características que se necesiten cubrir en el mismo, McConnell (1997) menciona que este modelo de cascada pura es muy

recomendable seguirlo cuando se tiene poco tiempo para desarrollar el sistema, ya que tiene etapas que en este modelo se tienen las definiciones de los requerimientos en forma estable, es decir, que no tienen cambios en el transcurso del desarrollo, por lo mismo de que no existen cambios en los requerimientos, no es necesario volver atrás en alguna actividad que ya hayamos trabajado en ello.

Como en este modelo, una de las principales desventajas como lo podemos apreciar en la figura 2.4, es que no se puede uno dar el lujo de regresar atrás ya que ocasiona un gasto mayor, debido a este inconveniente, "se tienen que especificar completamente todos los requerimientos al comienzo del proyecto" (McConnell, 1997).

2.3.3 Modelo de Ciclo de Vida Codificar y Corregir (Code and Fix)

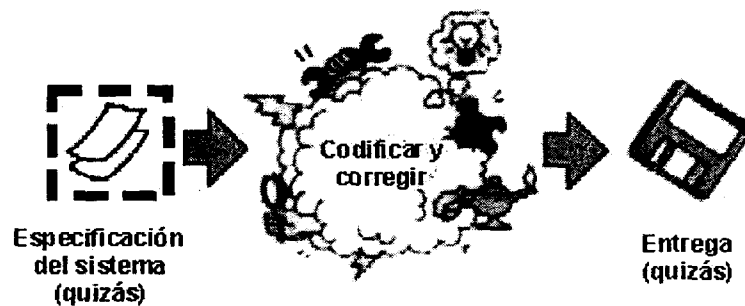


Fig. 2.5 Modelo de Codificar y Corregir (Code and Fix)
[Fuente: McConnell, 1997]

A muchas de las personas que desarrollan aplicaciones de software les parece que el llevar a cabo la codificación inmediata de los requerimientos del usuario es lo más conveniente y fácil de hacer, ya que se ahorran tiempo en muchos análisis que consideran no son necesarios llevarlos a cabo.

Las actividades de análisis y diseño, no se llevan a cabo antes de realizar la codificación si no que se da con el mismo desarrollo, McConnell (1997) nos dice que este modelo ofrece las siguientes ventajas:

- No conlleva ninguna gestión: planificación, documentación, control de calidad, cumplimiento de estándares
- Muestra indicios inmediatos de resultados

Se puede considerar que este tipo de modelo es muy peligroso, ya que no se puede pasar por alto algunos errores que pueden dar en verdad algunos dolores

de cabeza, más cuando tengamos que "desechar el trabajo y comenzar de nuevo" (McConnell, 1997).

2.3.4 Modelo de Ciclo de vida en Espiral

El modelo en espiral es un modelo de ciclo de vida orientado a riesgos que divide un proyecto de software en miniproyectos (McConnell, 1997). Al iniciar cada etapa, se ve como si fuera un miniproyecto, esto lleva a pensar que tal vez se pueda incluir algún otro modelo para desarrollar la etapa en la cuál se encuentra. Este es un modelo, que en realidad es algo complicado desarrollar, ya que se necesita una clara visión en cada etapa del mismo para no cometer errores que nos ocasionen desvíos importantes en el desarrollo del proyecto.

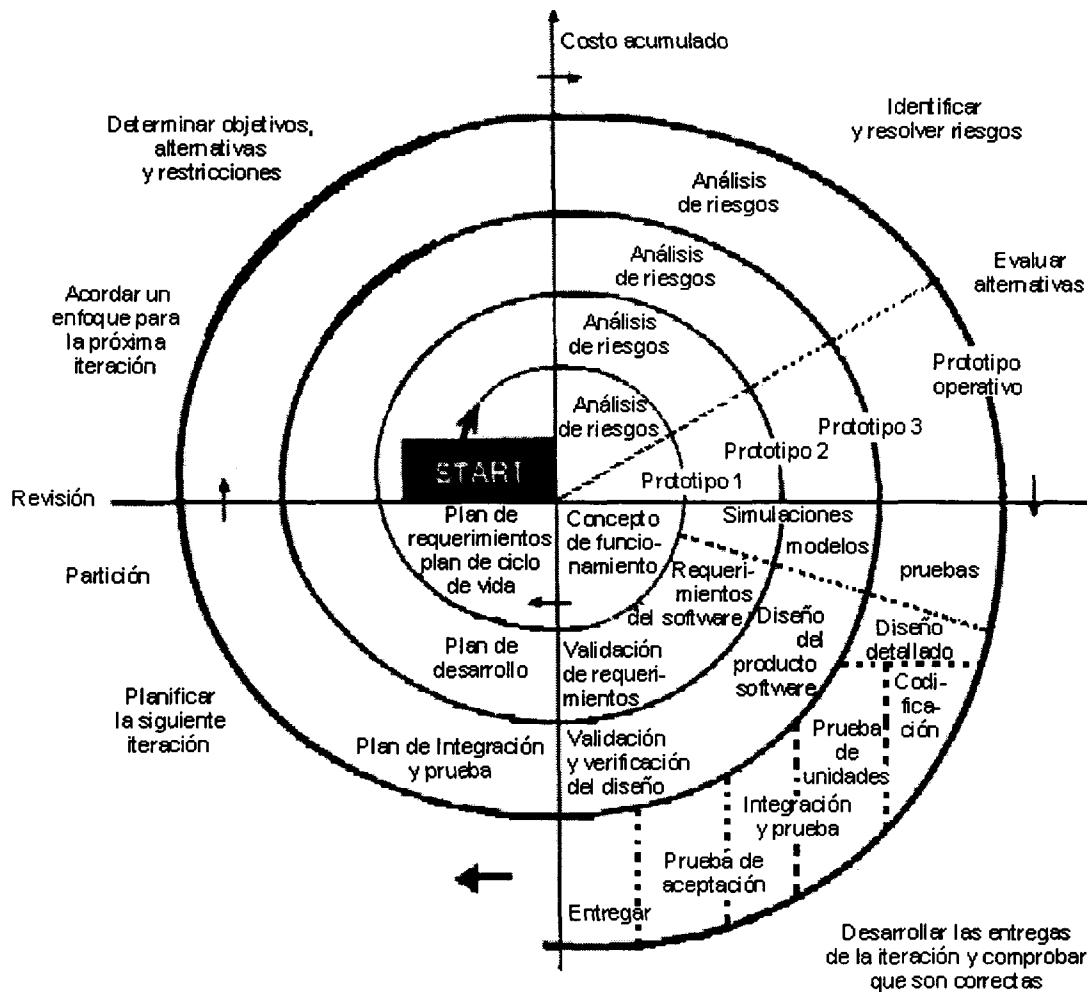


Fig. 2.6 Modelo en espiral [Fuente: McConnell, 1997]

Este modelo puede adaptarse a las necesidades particulares de cada proyecto, así entonces no significa que tenga que seguirse al pie de la letra lo que se muestra en la figura. Una variación a este modelo es presentada por King (1992), en donde dice que el desarrollo de un ciclo de vida de un sistema es muy cerrado a una espiral en donde se especifican un conjunto de eventos secuenciales. Esta variación comienza por el lado contrario del presentado por McConnell (1997), se inicia de afuera hacia dentro comenzando con la definición de requerimientos (que es la primer etapa en cada modelo) y termina con la liberación del sistema.

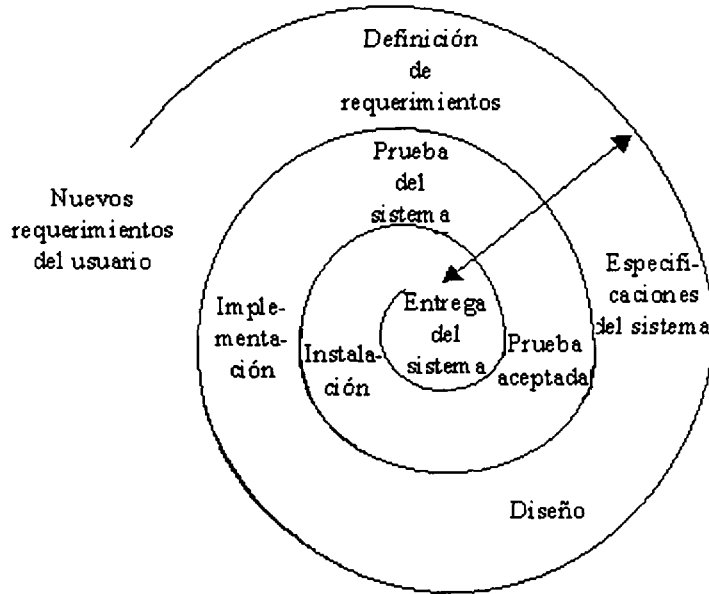


Fig. 2.7 Modelo en espiral simple
[Fuente: King, 1992]

2.3.5 Modelo de Ciclo de Vida de Cascadas Modificadas (Sashimi)

McConnell (1997) presenta una versión adaptada al software del modelo presentado por DeGrace (1990), en donde nos muestra en la fig. 2.8.

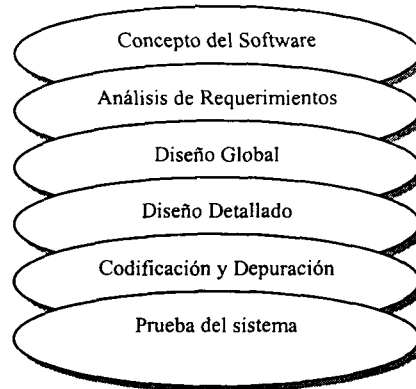


Fig. 2.8
Modelo de cascadas modificadas
[Fuente: McConnell, 1997]

En este modelo, se propone iniciar la etapa siguiente antes de que se termine la etapa actual en la que estamos trabajando. Esto puede traer problemas, ya que si no existe una adecuada comunicación entre los encargados de desarrollar cada una de las etapas, será un poco desgastante para el equipo de trabajo el desarrollar ambas fases en determinado momento.

2.3.6 Modelo de Ciclo de Vida de Cascadas Modificadas (Subproyectos)

Una opción más que se ubicaría perfectamente en proyectos en donde podemos desarrollar algunas actividades en paralelo es el Modelo en Cascadas Modificadas con Subproyectos.

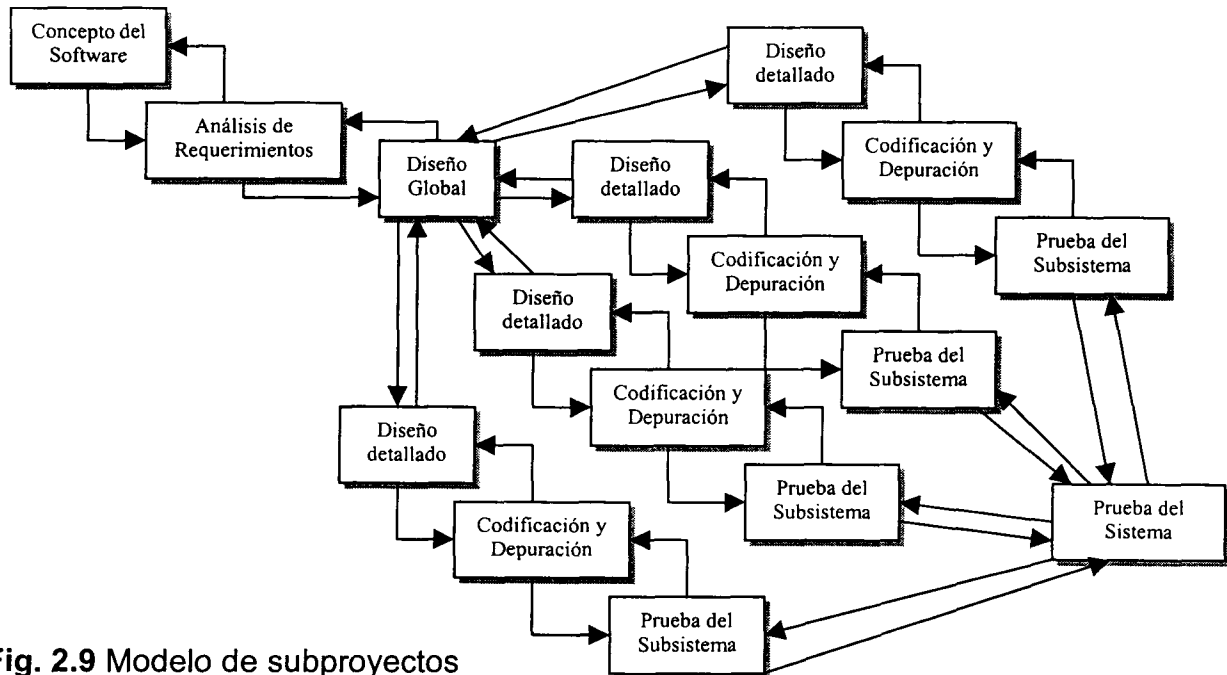


Fig. 2.9 Modelo de subproyectos
[Fuente: McConnell, 1997]

Este modelo se puede aplicar cuando al llegar a la etapa del diseño global, si las características del proyecto nos permiten subdividirlo en varios subproyectos, el diseño detallado de cada subproyecto puede llevarse a cabo al mismo tiempo, realizándolo entre varios equipos encargados cada uno de un subproyecto, así se pretende ahorrar sobretodo tiempo en el diseño, ya que se desarrolla todo el proyecto poco a poco según se avanza en el diseño de cada subproyecto.

2.3.7 Modelo de Ciclo de Vida Prototipo Evolutivo

Este modelo, ilustrado en la fig. 2.10, es similar al de Codificar y Corregir, a diferencia de que en el prototipo evolutivo real incluye análisis de requerimientos real, diseño real y código pensando para el mantenimiento real, en niveles ligeramente inferiores de los que se utilizan con las aproximaciones tradicionales (McConnell 1997).

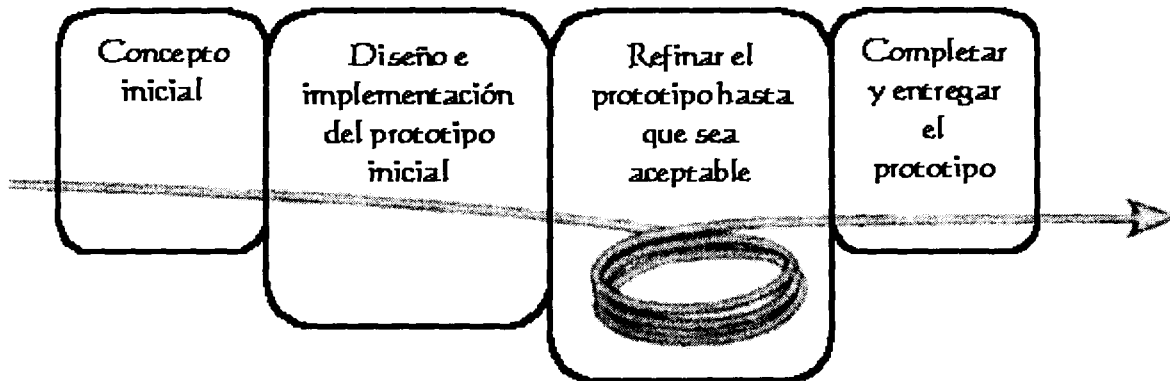


Fig. 2.10 Modelo de prototipo evolutivo
[Fuente: McConnell, 1997]

Como se menciona anteriormente es similar al de Codificar y Corregir en el punto de que los dos se desarrollan según el cliente va dando sus requerimientos, y se presenta más que nada cuando ni el cliente tiene bien definido que es lo que quiere como producto final, ni el desarrollador sabe interpretar o identificar las necesidades reales del cliente, o bien no se está seguro sobre que camino seguir en cuanto a la selección de algoritmos o arquitectura a utilizar (McConnell, 1997).

2.3.8 Modelo de Ciclo de Vida Entrega por Etapas

Este modelo, ilustrado en la fig. 2.11, tiene una semejanza tanto con el modelo de cascada como con el de prototipo evolutivo. A diferencia de los modelos mencionados anteriormente, es que éste tiene la oportunidad de ir entregando partes funcionales al 100% según se vaya desarrollando, ya que se termina el desarrollo de una parte del proyecto global, esta parte es entregada al cliente, debidamente probada y comprobada funcionando al 100%, y no como en el de cascada y prototipo evolutivo que se entregan una vez que se termino de desarrollar todo el producto (McConnell, 1997).

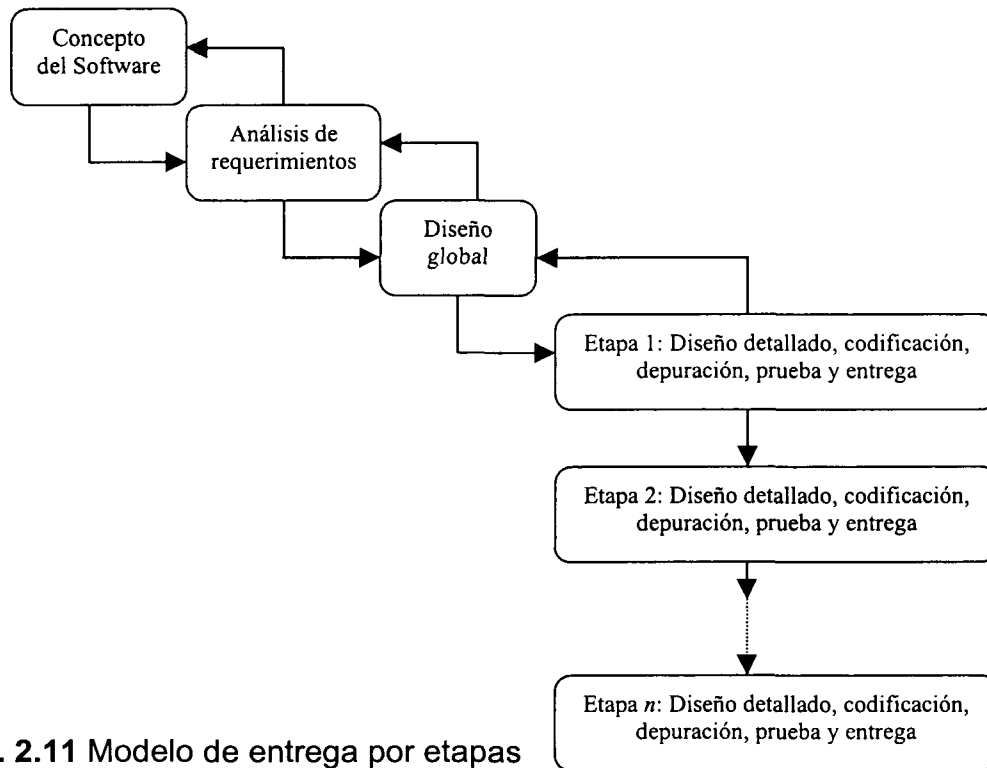


Fig. 2.11 Modelo de entrega por etapas
[Fuente: McConnell, 1997]

McConnell (1997) asegura que este modelo tiene la dificultad de que se desarrolle con éxito debido a que si existe una mala planificación en los niveles tanto técnicos como de gestión, no podrán asignarse tareas adecuadas al personal adecuado y éstos no podrán cumplir con los requerimientos y tiempos marcados por el líder del proyecto.

2.4 Conclusiones

Al desarrollar un proyecto, existen diferentes alternativas de llegar a la meta, siendo todas válidas. Puede ser que la alternativa 1 a seguir sea muy similar a la alternativa 2, pero existe alguna diferencia que la hace mejor o peor para determinado proyecto.

De la misma forma ocurre al visualizar un proyecto a desarrollar y tener la necesidad de seleccionar el modelo de Ciclo de Vida más adecuado al proyecto en sí. La selección de dicho modelo, estará sujeta a la naturaleza misma del proyecto, a la complejidad que este tenga, incluyendo factores como el económico, el tecnológico, humano, y sobretodo, el tiempo.

De la conjunción de todas las variables mencionadas, dependerá la selección del modelo a seleccionar, por ejemplo, si se requiere que en una semana se muestre un avance de proyecto, entonces no es válido ni recomendable seguir el modelo de Sashimi, ya que el factor tiempo marcaría una enorme presión en el equipo de trabajo, por lo que lo más recomendable sería tomar el modelo de Prototipo Evolutivo, que mostraría las funciones que se desarrollarán en el proyecto dando a los directivos un panorama general, y así tomar la decisión de hacerlo o cancelarlo. El Anexo 4 muestra una tabla comparativa de los modelos mencionados en este capítulo.

Capítulo III Administración de Proyectos

3.1 Introducción

El seleccionar un modelo de ciclo de vida es solo el inicio en la vida de un proyecto. Una vez que se ha determinado el modelo más adecuado a las características del proyecto, una buena práctica es analizar cuáles son los elementos que se incluirán en el trabajo (recursos tecnológicos, económicos y humanos) para determinar el plan de trabajo, estableciendo las responsabilidades y obligaciones de cada uno de los miembros del equipo.

Esto es estrictamente indispensable saberlo, ya que de lo contrario no se tendrá un control del avance del proyecto, no se podrá saber con qué recursos estamos trabajando bien y mucho menos saber si nos hace falta algo que no identificamos. De igual manera, el Líder o Administrador de Proyectos, debe tener en cuenta que en el transcurso del desarrollo, se presentarán algunos problemas que de una u otra manera deben ser resueltos.

En este capítulo se pretende dar al lector un panorama general de aquellos problemas que pueden presentarse en el desarrollo de los proyectos informáticos, con el firme propósito de que los tenga en cuenta no para corregirlos, si no más bien para tratar de evitar que se presenten.

3.2 ¿Qué es la Administración de Proyectos?

Hoy en día, y en cualquier área de trabajo, las funciones que se desarrollan para llevar a cabo alguna actividad, están intrínsecas a un proyecto, ya que todo trabajo tiene un inicio y un objetivo final. Para llegar al final del camino, es necesario trazar una ruta a seguir para poder cumplir nuestra meta, siendo el Administrador del Proyecto el responsable de trazar esa ruta a seguir, y la Administración de Proyectos la tarea fundamental para conseguir que se realicen de acuerdo a lo planeado.

Para tener una visión clara de que hablamos al referirnos a la Administración de Proyectos, se cita a continuación a varios autores, con el objetivo de dejar más claramente las funciones de este. Primero veremos una definición concerniente a Administración, y una referente a lo que es un proyecto.

La administración es un proceso de coordinación para la optimización de los recursos humanos y materiales de un organismo social, cuya finalidad es la de

incrementar el bienestar social. El proceso de administración implica llevar a cabo las siguientes etapas:

- Previsión
- Planeación
- Organización / Integración
- Dirección y control

Un proyecto es un proceso único con un principio y un fin bien definidos, que consiste en una serie de actividades interrelacionadas entre sí, que deben ejecutarse para lograr un objetivo predeterminado. La administración de proyectos puede definirse como la coordinación para la optimización de los recursos humanos y materiales de una serie de actividades relacionadas entre sí, que deberán ejecutarse para lograr el objetivo de un proyecto Field (1997).

Weiss y Wysocki (1992), nos dicen que cuando pensamos en los principios de la administración, usualmente lo asociamos con la administración de gente. Esos mismos principios aplican a los proyectos. Al pensar en esto, nos referimos a:

- Planeación: Involucra el establecimiento claro y preciso de los objetivos (así como las tareas a realizar en cada uno de ellos) en orden para alcanzar una meta final
- Organización: Además de organizar a la gente, la administración de proyectos incluye el reunir los recursos necesarios (gente, materiales y dinero) para llevar a cabo el trabajo, involucra crear la estructura necesaria para ejecutar el plan de trabajo.
- Control: una vez que los recursos son ensamblados en su estructura, es necesario monitorear y mantener la estructura en el progreso del proyecto, incluye la definición y definición de una estructura de reportes de puntos específicos a través de la vida del proyecto.
- Cambios: algunas situaciones una vez identificadas necesitan cambios, el administrador de proyectos es el indicado para realizarlos.

Ahora, una vez que sabemos qué es Administración y qué es un Proyecto, veremos lo que es una Administración de Proyectos. Comenzando con la definición de Ainsworth (1999), al hablar de Administración de Proyectos, estamos hablando de que éste, es visto como "un método o conjunto de técnicas basadas en los principios de la administración usados para planear, estimar y controlar actividades de trabajo para alcanzar un resultado final deseado en un tiempo determinado, dentro de un presupuesto y acorde a las especificaciones".

Como se puede ver, en el concepto de Ainsworth (1999) no se hace referencia explícita a un proyecto de informática, mas bien es una definición general la cuál se enfoca hacia las actividades que se deben de realizar, sea cual sea la naturaleza del proyecto.

Por otro lado, King (1992) dice que "la administración de proyectos es una disciplina que proporciona un gran servicio a la industria de sistemas de computación, pero no siempre es tomada en cuenta en las batallas del desarrollo de sistemas".

¿Por qué la administración de proyectos?

El propósito básico al iniciar un proyecto es acercarse a algunas metas de la organización. "La razón de organizar tareas como un proyecto, es enfocar la responsabilidad y autoridad para el logro de metas en un individuo o grupo pequeño", Meredith y Mantel (1989).

3.3 Características de los Proyectos de Informática

Antes de revisar cuales son las características que debe de cumplir un proyecto, cabe recordar que anteriormente se dijo que todos los proyectos son diferentes de acuerdo a su origen y destino que buscan, es por eso que no se puede dar una lista de características definitivas para clasificar a los proyectos. Por un lado, Pereña (1991) nos muestra que un proyecto se identifica por un conjunto de notas características que lo diferencian de otras actividades de tipo continuo que se realizan en la empresa:

- Entidad, tamaño, alcance
- Medios importantes, variados y cambiantes
- Discontinuidad
- Dinamismo y evolución
- Irreversible
- Influencias externas
- Riesgo

Así mismo, Weiss y Wysocki (1992) nos dicen que un proyecto es aquel que tiene las siguientes características:

- Complejas y numerosas actividades
- Unico, con plan de actividades
- Finito, con un inicio y un fin
- Recursos y presupuesto limite

- Mucha gente involucrada, usualmente a través de las áreas funcionales de la organización
- Actividades secuenciales
- Orientado a metas
- Debe dar como resultado un producto final

Como podemos apreciar y de acuerdo a lo mencionado anteriormente, un proyecto nunca será igual a otro, ya que cada uno tiene sus propias características que lo hace diferente, aunque no se puede decir que no existan algunos que sea parecidos.

Pereña (1991) nos dice que si pensamos que una correcta gestión de los proyectos, aunque no permita garantizar que todos y cada uno de ellos alcance plenamente sus objetivos, debe producir en la mayor parte de los casos, que la operación culmine con resultados similares a los apetecidos. Es lógico que en ciertas ocasiones, un proyecto pueda topar con dificultades impensables o con obstáculos imprevisibles.

El desarrollo de un proyecto no es nada sencillo, y si no se tiene el conocimiento sobre como llevar a cabo una adecuada planeación y control de actividades, el proyecto puede ser que nunca se llegue a terminar, y por consiguiente se tendría un fracaso en el mismo. Los departamentos de sistemas suelen dedicar mucho tiempo y recursos a resolver las necesidades de todo el mundo dentro de la empresa, excepto las suyas propias. Cuando de cubrir sus propios requerimientos se trata, la regla es el "bomberazo" (De la Torre, 1998).

3.4 Antes de comenzar un Proyecto

Los proyectos informáticos deben de estar visualizados dentro del plan de la organización, aunque no todos puedan llevarse a cabo. A pesar de que se sabe que la informática debe de existir para apoyar a la organización, que es importante y no tanto por el valor que pueda tener por sí misma sino por los beneficios que se le pueda brindar a esta. Así de esta manera es necesario hacer todo lo posible por apoyar los programas y planes de la organización a la que servimos Alanís (1998).

De tal manera, que antes de poder establecer un plan de desarrollo para un proyecto informático, Souder citado por Meredith y Mantel (1989) sugiere que debe de realizarse una evaluación del modelo a seguir para determinar si cubre los siguientes puntos:

1. Realista. El modelo a seguir y el proyecto deben reflejar la realidad de las decisiones del administrador incluyendo objetivos múltiples tanto para la firma como para los administradores.

2. Capacidad. El modelo debe de ser lo suficientemente sofisticado para distribuirse en diferentes periodos de tiempo, simular situaciones tanto internas como externas al proyecto y optimizar la toma de decisiones.
3. Flexible. El modelo debe de dar resultados válidos dentro del rango de condiciones establecidas por la empresa. Debe dar la habilidad para poder modificarlos de acuerdo al ambiente de la firma.
4. Fácil de usar. El modelo debe de ser razonablemente conveniente, no tomar mucho tiempo para llevarlo a cabo y fácil de usar y comprender.
5. Costo. La recolección de datos y modelación de los costos deben ser mucho menores a los costos del proyecto.

Así mismo, aparte de considerar los factores mencionados anteriormente, deben dar prioridad a los proyectos, ya que no todos los proyectos informáticos que se lleven a cabo tienen el mismo peso e importancia para la organización. Alanís (1998) nos dice que el orden que se le ponga a los proyectos va a depender de cuatro factores principales:

- ¿qué tan importante es el programa al que se está apoyando?
- ¿qué tan importante es el apoyo informático para el proyecto?
- ¿qué tan disponible está la tecnología?
- ¿qué disposición tienen los usuarios?

De tal manera, es necesario resaltar dentro del análisis previo para realizar un proyecto, que lo más importante es tener en mente que se busca la optimización de los recursos humanos, materiales y tecnológicos para ayudar a la organización a hacer un mejor trabajo, y de este enfoque depende en gran medida el posible éxito del proyecto.

3.5 ¿Por qué fracasan los Proyectos de Informática?

El conseguir las respuestas para resolver las fallas de los proyectos usualmente recae en el desarrollo de una comunicación escrita como las definiciones del proyecto, planes del proyecto y detalle de especificaciones. De cualquier manera, uno de los mayores problemas con cualquier comunicación escrita es el nivel de entendimiento de los lectores.

Como gente de tecnología, pensamos, escribimos, y hablamos de una manera que no es realmente comprensible para la gente fuera de la industria (Standish, 2000).

Los sistemas de información están condenados al fracaso debido a que los desarrolladores fallan en identificar las necesidades del usuario o definir el rumbo exacto del proyecto, o peor aún, ambos (Field 1997).

Para ejemplificar lo mencionado anteriormente, podemos apreciar en la **Fig. 3.1**, el porcentaje mayor de causas de error, son precisamente el no saber determinar las necesidades reales del usuario, y no otros factores como la habilidad o conocimientos del equipo de trabajo por mencionar alguno.

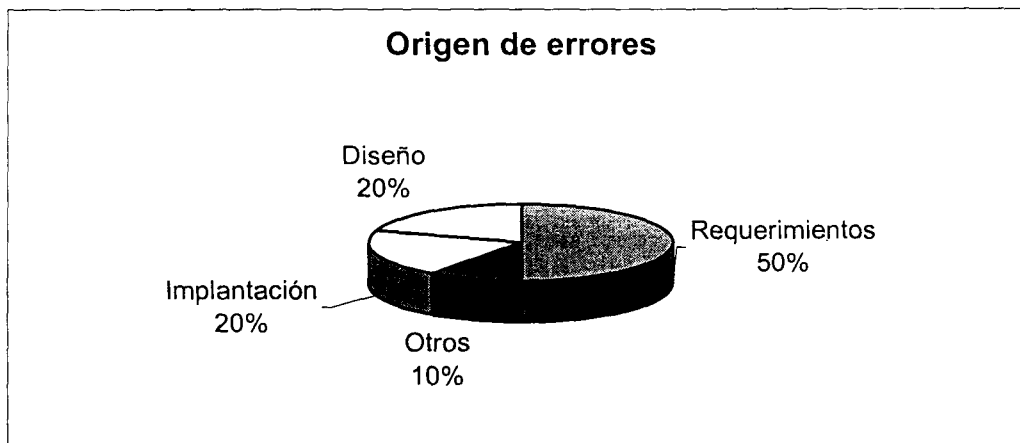


Fig. 3.1 Origen de errores

Fuente [Martínez, 1996]

Al iniciar los trabajos de un nuevo proyecto, todos los miembros del equipo comienzan a trabajar con un gran entusiasmo para participar en el desarrollo. Desafortunadamente no todo es color de rosa. Los proyectos están sujetos a una especie de maldición por lo cuál es imposible que se cumplan los objetivos iniciales. En muchos casos los objetivos se cumplen pero con plazos que frecuentemente triplican el objetivo inicial (Pereña, 1991).

3.6 Indicadores de falla en los Proyectos de Informática

Algunas prácticas inefectivas de desarrollo son seleccionadas a menudo por mucha gente, obteniendo los predecibles malos resultados, a los que se les conoce como "errores clásicos". La mayoría de estos errores tiene un cierto atractivo. ¿Necesitas rescatar un proyecto que está atrasado?, Añade más gente al equipo. ¿Quieres reducir tu calendarización? ¡Acorta las actividades!. ¿Es uno de tus elementos clave quien esta afectando a todo el equipo? Espera a terminar el proyecto y despídelo, ¿necesitas que el proyecto funcione en su totalidad? Toma a todos los programadores disponibles y comienza el proyecto lo más pronto posible McConnell (2000).

En muchas ocasiones, la “administración” de la tecnología es la causa limitante de que los proyectos de sistemas de información más que la tecnología por sí misma, tengan en general más fallas que éxitos.

Un gran número de investigadores e instituciones en todo el mundo han estudiado el porqué de las fallas de los sistemas. Flowers citado por Liebowitz (1999) identificó los siguientes factores críticos en fallas de sistemas de información:

- Temor
- Estructura de reportes deficiente
- Consultoría deficiente
- Sobrecomité
- Cambios de requerimientos
- Presiones políticas
- Consecución débil
- Enfoque tecnológico
- Desarrollar en situaciones agrietadas
- Complejidad subestimada
- Pruebas inadecuadas
- Entrenamiento deficiente

Si revisamos alguno de los modelos de ciclo de vida de los sistemas de información, y analizamos a profundidad los factores anteriormente mencionados, se podrá encontrar que existen factores externos al ciclo de vida que provocan las fallas del desarrollo, como pueden ser las presiones políticas, por mencionar un ejemplo.

Dicho anteriormente que los proyectos no son únicos, y que las características de los mismos varían de acuerdo a la naturaleza y enfoque del proyecto, encontramos que las causas o indicadores de problemas en el desarrollo del proyecto no pueden ser la excepción. Valdéz (1997) nos presenta una lista de algunos de los errores más comunes que existen en el desarrollo de proyectos de software, los cuales se presentan a continuación:

- No existe una metodología
- No existen métricas de medición y control de avance
- La metodología carece de procedimientos concretos
- Apego muy estricto a la metodología
- Carencia de modelos apropiados
- Uso inapropiado o erróneo de las herramientas de modelación
- Desarrollo inapropiado

Como se puede apreciar, es un poco complicado tener un adecuado desarrollo si no se consideran los puntos anteriormente mencionados. Estos puntos presentados se enfocan más que nada al proceso de desarrollo, es decir, cuando el desarrollo del sistema está en marcha, aunque no son todos los posibles puntos de conflicto que originan que el proyecto se desvíe y en muchas ocasiones llegue al fracaso.

Por otro lado, Field (1997), al igual que Valdéz (1997) nos muestra lo que a su experiencia son causas de fallas en el desarrollo de sistemas de información:

- Los administradores de proyecto no entienden las necesidades del usuario
- Mala definición del ambiente
- Cambios en el proyecto son pobremente manejados
- Cambios en la elección de la tecnología
- Cambios en las necesidades del negocio
- Fechas irreales
- Resistencia de los usuarios
- Pérdida del patrocinio
- Carencia de gente con habilidades
- Las mejores prácticas y lecciones aprendidas son ignoradas

Estas causas presentadas por Field (1997), son un poco diferentes a las ilustradas por Valdéz (1997), aunque en esta ocasión se identifica que algunas se encuentran en la etapa inicial (definición de requerimientos del usuario) así como en otras etapas posteriores como el diseño del proyecto.

Si se continúa analizando posibles causas de errores en el desarrollo de sistemas, podremos encontrar otras muchas más, como las que nos presenta McConnell (1999a), quien nos presenta una lista de algunos errores llamados "clásicos" considerados por el cómo las prácticas erróneas en el desarrollo de software, los cuales llevan a tener malos resultados. Los "errores clásicos" que McConnell presenta son:

- Motivación indeterminada
- Oficinas ruidosas
- Abandono del plan bajo presión
- Eliminar actividades hacia arriba
- Eliminar pruebas de calidad para improvisar desarrollo rápido
- Carencia de control de cambios futuros (*Lack of feature-creep control*)
- Gastar demasiado tiempo en el frontal difuso (*fuzzy front end*)

Por su parte, Pereña (1991) nos dice que la mayoría de los proyectos de sistemas de información terminan por que se presenta alguna o en su defecto, varios de los siguientes factores:

- El cliente no existe o no está plenamente identificado
- Existen varios clientes simultáneos con objetivos o preocupaciones diferentes
- El cliente no tiene las ideas claras y titubea, o se vuelve atrás de decisiones anteriores
- El cliente cambia durante la vida del proyecto

Standish Group y Clegg, citados por Liebowitz (1999) también muestran algunas cifras (olvidándonos un poco de indicadores de fracaso) en las que se encuentran los proyectos de sistemas de información:

Por ejemplo, Standish Group citado por Liebowitz (1999) muestra los siguientes porcentajes:

- 31% de nuevos proyectos de Sistemas de Información son cancelados antes de completarse con un costo estimado combinado de \$81 billones de dólares
- 52.7% de los proyectos terminados están con un 189% sobre el presupuesto adicional al costo de \$59 billones.

y por lo que corresponde a su parte, Clegg quien también es citado por Liebowitz (1999) encontró que:

- arriba del 90% de todos los proyectos de Tecnologías de Información fallan al conocer sus metas
- 80% son entregados fuera de tiempo y sobrepresupuestados y
- 40% son abandonados

Esto nos da una muestra del por qué es importante saber, en donde pueden encontrarse las causas de que el desarrollo de uno de nuestros proyectos esté fallando, o mejor aún, el tratar de prevenir que los errores o señales que se han mencionado se puedan presentar en el ciclo de vida del mismo, tomando las acciones preventivas que sean necesarias. "Las señales de precaución de falla son detectables a lo largo del proyecto. Solo tienes que saber en donde mirar", Field (1997).

Los proyectos fallan o mueren simplemente debido a que el departamento de Sistemas de Información falla al seguir los principios básicos de la administración de proyectos que ayudan a asegurar el éxito en la ingeniería e industria de la construcción. De acuerdo a lo mencionado por Jones quien es citado por Field (1997) existen muchas excusas acerca de como las nuevas tecnologías son inestables o las organizaciones de Sistemas de Información son cambiantes, esta es la razón principal por la cuál, considera que el principal culpable del fracaso de los proyectos es el Administrador de Proyectos.

Para recaer la culpabilidad aún más en el responsable de informática de las organizaciones, Gopal K. Kapur, presidente del *Center for Project Management* (CPM) ubicado en San Ramón, Calif., quien también es citado por Field (1997),

nos dice que "el CIO es una de las mayores razones para las fallas de los proyectos".

Retornando un paso atrás a las cifras descritas en los párrafos anteriores, es conveniente revisar también otros indicadores que nos muestran otros autores. Por ejemplo, Ewusi-Mensah citado por Liebowitz (1999) indica que la cancelación de proyectos de Sistemas de Información es usualmente una combinación de

- Metas débiles
- Conformación de equipo de proyecto deficiente
- Carencia de administración y control del proyecto
- Pocos conocimientos técnicos
- Deficiente infraestructura tecnológica
- Carencia de involucramiento del administrador
- Subir los costos y tiempo de complementación del proyecto

Al revisar esta última lista, se puede observar que una gran cantidad de factores se encuentran localizadas en la parte inicial del proyecto, y revisten su importancia en el sentido de que si no son plenamente definidos e identificados, las etapas subsecuentes tendrán problemas, recordando el dicho, "el que mal empieza, mal acaba".

Los proyectos no tienen problemas únicamente con las habilidades de los elementos del equipo de trabajo, se presentan problemas tanto con el personal, los procesos, la tecnología y el producto en sí, tal como lo muestra McConnell (2000) en la tabla 3.1, en la cuál nos muestra, aquellos errores considerados clásicos, clasificados de acuerdo al factor en el cuál se presenta dicho error.

Tabla 3.1 Sumario de errores clásicos
[Fuente: McConnell, 2000]

Errores relacionados a la gente	Errores relacionados al proceso	Errores relacionados al producto	Errores relacionados a la tecnología
Motivación indeterminada	Calendarización sobrepasada	Requerimientos enchapados	Síndrome Silver-bullet
Personal sin habilidades	Administración de riesgos deficiente	Arrastre futuro	Sobreestimación para nuevas herramientas o métodos
Problemas incontrolables con el equipo	Contratación inadecuada	Desarrollo enchapado	Intercambiando herramientas a la mitad del proyecto
Héroes	Abandono del plan por presión	Empujar a la negociación	Carencia de control del código
Ajustar el proyecto a gente que se incorpora tarde	Gastar demasiado tiempo en la parte de atrás del sistema	Desarrollo orientado a la investigación	
Oficinas ruidosas	Acortamiento de actividades		
Fricción entre clientes y desarrolladores	Diseño inadecuado		
Expectativas irrealistas	Acortamiento de proceso de calidad		
Falta de patrocinio directivo	Insuficientes controles administrativos		
Carencia de cliente potencial	Convergencia prematura		
Falta de usuarios	Omitir tareas		
Sobreponer políticas a resultados	Planeación sobrepasada		
Supuestos	Codificación sin diseño		

Bienkowski (1989) hace referencia a una metodología de trabajo, resaltando que los proyectos que no siguen una metodología usualmente caen en los siguientes errores:

- El proyecto es una solución en busca de un problema
- Únicamente el equipo del proyecto está interesado en el resultado final
- Nadie está a cargo
- El plan del proyecto carece de estructura
- El plan del proyecto carece de detalle
- El proyecto está mal presupuestado
- Tiene insuficientes recursos
- El proyecto no va de acuerdo a lo planeado
- El equipo del proyecto no tiene comunicación
- El proyecto está alejado de las metas originales

Como se puede apreciar, cada uno de los autores citados, encuentra indicadores semejantes, aunque no se muestra una repetición continua de algún indicador entre un autor y otro. Esto lleva a pensar, que al trabajar en un proyecto, además de todos los indicadores mostrados anteriormente, podríamos encontrarnos con algún otro que ni siquiera imaginemos. Se podría llegar a concluir que los daños y fallas a los proyectos podrían atribuirse a una inadecuada implementación de tecnología nueva o inmadura, pero ésta, sólo representa el 5% de las fallas de proyectos Biggs (2000).

Las presiones políticas por parte de los directivos, son un clásico ejemplo que origina que algún proyecto se cancele o se retrase. McConnell (2000a) presenta un caso de estudio sucedido en la empresa Giga Safe. En dicho caso, se desarrollaría un proyecto propuesto y presentado por uno de sus programadores, el cuál buscaba automatizar los procesos operativos de la empresa. El mismo programador sería el líder del proyecto, y había estimado que el proyecto se desarrollaría en 12 meses. Los directivos apoyaron el proyecto pero debería de ser desarrollado en no más de 6 meses. Finalmente el proyecto se llevó 13 meses de desarrollo, contando en ese íter con varios problemas ocasionados por las presiones de los directivos, y teniendo un costo mucho más elevado al inicialmente presupuestado.

3.7 Conclusiones

Se dijo que no todos los proyectos son iguales en sus características, por lo tanto esto nos lleva a pensar que cada proyecto se debe de atacar de diferente manera para poder cumplir con las metas establecidas. Así como no son iguales

en sus dimensiones, alcances demás características, también varían los problemas que se presentan en cada uno de ellos.

Los problemas no son únicos para determinado tipo de proyecto. En cada uno de los proyectos que desarrollemos están latentes todas las posibilidades de error o problema que se presentaron en este capítulo, por lo tanto es necesario tomarlas en cuenta no solo a unas cuantas, si no mas bien a todas con el propósito de estar alertas para tratar de evitar que estos se presenten, y en su defecto, identificarlos a tiempo para hacer que cause el menor daño posible al desarrollo del proyecto.

El anexo 5 muestra un mapa de problemas que se presentan en el ambiente de los proyectos. Muestra tres elementos principales a considerar cuando se realiza un proyecto:

1. Usuario (cliente)
2. Líder del Proyecto
3. Equipo de Trabajo

Estos son los tres elementos que hemos manejado en el desarrollo de la tesis. El anexo 5 muestra los diferentes indicadores que presentan cada uno de estos elementos, y que se conjugan en el núcleo del proyecto, afectando diversos factores como:

- Unión del equipo de trabajo
- Presupuesto
- Calendario de actividades
- Alternativas de desarrollo entre otros.

Esta mezcla e invasión de problemas vuelven tensas las relaciones entre sus participantes, haciendo un ambiente no adecuado para el trabajo, ocasionando que el proyecto que se entrega al usuario, no esté acorde a las necesidades y requerimientos del usuario (visto como cliente, ya sea interno o externo a la empresa en donde se trabaja el proyecto), provocando que finalmente todo el trabajo que se haya realizado, simplemente sea desechado por no tener la calidad y no cubrir los puntos dispuestos al inicio del mismo. Podemos ver entonces que no únicamente el equipo de desarrollo es quién puede provocar que un proyecto no se lleve a cabo correctamente, sino que tanto el usuario y por supuesto, el Líder del Proyecto tienen su responsabilidad en el éxito o fracaso de cualquier proyecto que se desarrolle.

Capítulo IV Equipo de Trabajo

4.1 Introducción

Todo trabajo necesita ser encabezado por una persona, y ésta tener a varias personas a su cargo para poder desarrollar todas las tareas que se involucran en el desarrollo del mismo. Un Líder o Administrador de Proyectos, tiene a su cargo a un grupo de trabajo que se encarga de hacer toda la parte de desarrollo, entre ellos se puede tener analistas, diseñadores, programadores, Web Masters, etc., y el Administrador de Proyectos tiene entre sus responsabilidades el proporcionarles sus tareas y el darles las herramientas y facilidades necesarias para que su equipo de trabajo desarrolle todo su potencial en las tareas encomendadas.

Pero no solamente cuenta el equipo de desarrollo en un proyecto de informática, una personalidad que es importante y que no se toma en cuenta en muchas ocasiones es quien ordena o paga el proyecto, en términos más exactos estamos hablando del cliente. Contrario a lo que se pudiera pensar que el cliente no tiene responsabilidades en el éxito o fracaso de un proyecto, éste reviste de gran importancia al mismo ya que de él nació la petición y es el único que sabe exactamente que es lo que desea como producto final.

Una de las labores más importantes del Administrador de Proyectos es identificar las verdaderas necesidades del cliente, y plasmarlas en su plan de trabajo, ya que si no logra saber qué es lo que realmente desea y necesita el cliente, el trabajo que se desarrolle posteriormente, no será el adecuado. Entonces, todos y cada uno de los elementos que participan en el proyecto, tienen sus obligaciones.

4.2 El Cliente

Tal vez pudiera pensarse que el cliente no es una figura importante o trascendental dentro del ciclo de vida de un proyecto de informática. Mucho menos pudiera llegar a pensarse que el cliente tiene parte de responsabilidad para que un proyecto llegue a desarrollarse completamente y con éxito. Ciertamente el cliente no es el responsable directo del proyecto, ya que ésta responsabilidad se le atribuye única y exclusivamente al Administrador de Proyectos tal como lo señala Weiss y Wysocki (1992).

Contrario a que se pudiera pensar sobre la importancia del cliente en todo proyecto, el cliente forma parte del ciclo de vida del mismo. De acuerdo a lo que

dice McConnell (1997), el cliente es el destinatario final y es quien solicita el desarrollo del proyecto de acuerdo a sus necesidades muy particulares. En ocasiones, el desarrollo de un proyecto de software, se realiza sin tener un enfoque hacia el cliente, pensando que el cliente no sabe que es lo que realmente necesita y que el equipo de desarrollo es quien sabe realmente que es lo que necesita el cliente, sea este interno o externo.

La orientación al cliente, en opinión del mismo McConnell (1997), parece que puede ser demasiado intangible para tener un efecto claro en la velocidad de desarrollo. Lo que es muy cierto, es que intangible o no, las empresas que convierten las relaciones con los clientes en un asunto de la máxima prioridad han hecho que desaparezcan muchos de sus problemas de desarrollo, incluyendo el problema del desarrollo lento.

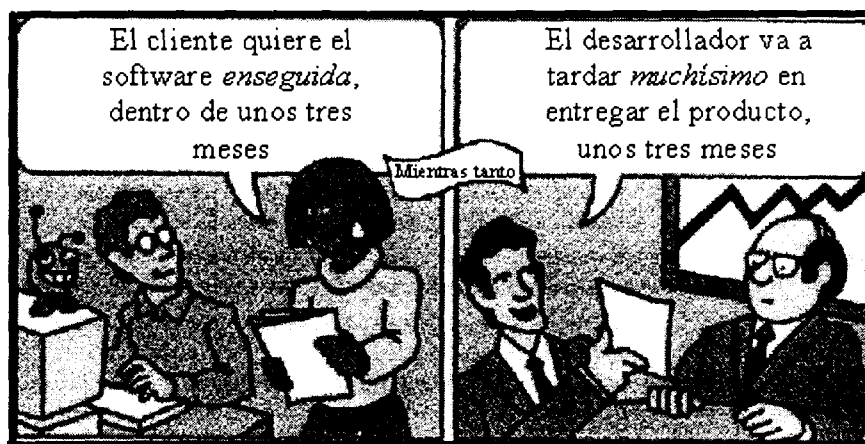


Fig. 4.1 Puntos de vista de un fenómeno.
[Fuente: McConnell, 1997]

La figura 4.1 nos muestra un ejemplo muy sencillo en cuanto a las diferentes perspectivas en cuanto al tiempo de desarrollo se refiere respecto a un proyecto determinado. La necesidad de prestar atención a sus clientes, se vuelve obvia cuando se da cuenta de que, al final, la sensación de sus clientes de si el desarrollo es lento o rápido es todo lo que importa. Si a los clientes no les gusta su producto, no lo querrán pagar, y si no quieren pagarlo, no hay nada más que puede importarle. Los ejecutivos se preocupan de la velocidad de desarrollo porque piensan que los clientes se preocupan por ello. Si puede contentar a los clientes, puede contentar a su jefe o a cualquier otra persona.

Es necesario hacer notar que el cliente varía mucho de proyecto a proyecto, esto no indica que la forma de tratarlo puede ser diferente. No es lo mismo un cliente que paga una cantidad grande por el desarrollo de una aplicación a un cliente que solo requiere de una actualización, o a un cliente interno, el cuál directamente no cubre los costos del proyecto si no que son cubiertos por la empresa. Independientemente de quién sea el cliente, y quién pague el proyecto, es conveniente que se piense al usuario final como el cliente, ya que

en muchas ocasiones, pensamos que el cliente es la parte directiva de la empresa y no el usuario final que es quién realmente utilizará en proyecto.

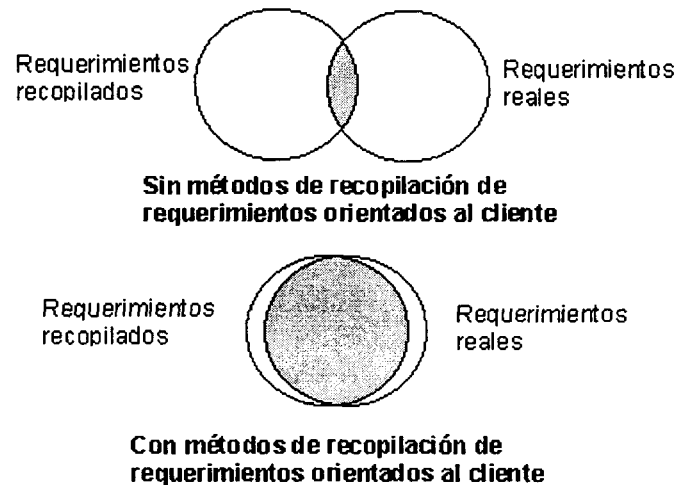


Fig. 4.2 Métodos típicos de recopilación de requerimientos y orientados al cliente.

[Fuente: McConnell, 1997]

La fig. 4.2 ilustra claramente lo que puede llegar a suceder si no se toma en cuenta como debe de ser al cliente en la determinación de los requerimientos, parte fundamental para el desarrollo de cualquier proyecto. Podemos apreciar que los métodos orientados al cliente aumentan la proporción de requerimientos reales que puede recopilar. Si los requerimientos no se llegan a determinar al 100%, simplemente todo plan de trabajo estará enfocado a un mal resultado. Si aún en estos momentos seguimos pensando, que el cliente o usuario final no importan mucho en el desarrollo del proyecto, basta con ver lo que dice el *Standish Group* que es citado por McConnell (1997), en el cuál, indica que tres de las razones principales para que los proyectos terminen tarde, sobrepasen el presupuesto y tengan una funcionalidad inferior a la deseada, son una deficiencia en la participación del usuario, especificaciones de requerimientos incompletos y modificación de los requerimientos y especificaciones.

Toda cancelación o suspensión de proyectos, repercuten finalmente en la economía de la organización, en el caso de que el cliente sea la empresa para la cuál trabajamos y el proyecto haya sido autorizado para algún departamento o usuario determinado, pero también repercute en el cliente externo, aunque en este punto tal vez el grupo de desarrollo no lo vea tan directo a como se vería dentro de una empresa.

La mejor forma de trabajar con el cliente y de evitar estos problemas, es utilizar algún método de trabajo que sea orientado a satisfacer en todo lo posible al cliente, de esta manera podemos obtener de acuerdo con McConnell (1997):

- Mejora de la eficiencia: la relación con el cliente puede estar en el camino crítico de un proyecto de software personalizado.
- Menos trabajo repetido: ya que es uno de los errores más costosos en el desarrollo de proyectos
- Reducción de riesgos: reducir los riesgos de que los clientes no sepan lo que quieren en realidad, que no se comprometan a un conjunto de requerimientos escritos, que no dejen realizar el trabajo de la gente, etc.
- Ausencia de fricción: puede nacer por parte del desarrollador o del cliente por no tener una adecuada relación entre ambos

La justificación para que se desarrolle un proyecto enfocado al cliente, es que las buenas relaciones con los clientes aumentan la velocidad de desarrollo y mejoran la percepción de la misma.

4.3 El Líder del Proyecto

Ya se a dicho que el único responsable directo en el éxito o fracaso de un proyecto de informática, es el Líder o Administrador del Proyecto, ya que es el responsable de la selección del equipo de trabajo, de asignar las tareas a los mismos y de realizar el plan de trabajo para llegar a terminar el trabajo a tiempo y asegurarse de que se a realizado para cumplir con los requerimientos establecidos anteriormente en el inicio del trabajo. Un Administrador o Líder de Proyectos no tiene exactamente las mismas funciones que cualquier otro Administrador de cualquiera de los demás departamentos dentro de la organización.

Meredith y Mantel (1989) marcan algunas diferencias entre un Administrador Funcional y un Administrador de Proyectos de Informática. Los departamentos diversos que existen en una organización, son encabezados usualmente por especialistas en el área, orientados y con conocimientos de los detalles de cada operación en la cuál ellos son responsables. Cuando existe una tarea técnicamente difícil, ellos saben como analizarla y resolverla. Como administradores funcionales, son los responsables para decidir cómo harán las cosas, quién las hará y que recursos deberán de asignarse a esa tarea.

Por otro lado un Administrador de Proyectos es usualmente un generalista con un amplio respaldo de experiencia y conocimientos. Un Administrador de Proyectos debe ver varias áreas, cada una con su propio especialista como se ilustra en la Fig. 4.3, en ella se muestran sus responsabilidades.

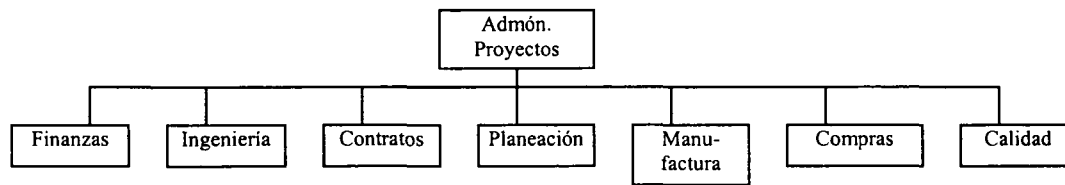


Fig. 4.3 Organización de una Administración de Proyectos.
[Fuente: Meredith y Mantel, 1989].

Por lo tanto, el Administrador de Proyectos requiere una habilidad para manejar varias piezas en una tarea unidas para formar una sola unidad. Esto es, el Administrador de Proyectos debe tener las habilidades de síntesis, mientras el Administrador Funcional debe tener más la habilidad de análisis.

Esta comparación entre el Administrador Funcional y el Administrador de Proyectos, revela otras cruciales diferencias entre los dos. El Administrador Funcional es un supervisor técnico directo. El Administrador de Proyectos es un facilitador. Conociendo la tecnología, el AF (Administrador Funcional) tiene el conocimiento básico requerido para supervisar y asesorar a sus subordinados de la mejor manera para manejar su trabajo y resolver los problemas en el curso normal del trabajo. El AP (Administrador de Proyectos) deberá tener los conocimientos técnicos más detalladamente en una o dos áreas específicas, pero ocasionalmente tiene conocimientos más profundos en esas áreas. Esta distinción entre facilitador y especialista es el elemento clave en la decisión de usar generalistas como AP más que especialistas.

El AP, no solamente necesita de conocimientos y una visión amplia sobre varias especialidades, además de esto, "un buen Administrador de Proyectos es la herramienta más importante para una exitosa entrega de un complejo proyecto de TI, pero incluso el más erudito administrador necesita más que experiencia y buenos presentimientos o impresiones para cerrar la implementación de proyectos de TI exitosamente" Radosevich (1999).

Si bien son necesarias ciertas cualidades para poder ser un Administrador de Proyectos, Weiss y Wysocki (1992) dicen que cualquiera de nosotros, puede llegar a ser uno de ellos, pero no cualquiera puede o tiene la capacidad de planearlo y administrarlo correctamente. Esto significa que para que un Administrador de Proyectos se digne de llamarse así, debe de ser alguien que tenga los conocimientos necesarios para saber qué etapas debe de cubrir para cumplir con los requerimientos del cliente, y estamos hablando que un cliente no necesariamente es alguien externo a la empresa, un cliente puede ser tanto interno (Contabilidad, Finanzas, Recursos Humanos, Dirección General, etc.) como externo (asesorías y consultoría a empresas, desarrollo de un producto, etc. enfocados a diversas entidades que no corresponden a la empresa).

Algunos líderes de proyecto pueden darse por vencidos fácilmente cuando ven que la recuperación de un proyecto que está a punto de perderse es muy difícil y optan por no realizar algún intento por salvarlo. Otros en cambio, reconocen que el proyecto está mal en algún punto, y se enfocan en resolver dicho problema. Field, Pearson y Schuneider (2000) presentan 5 casos de proyectos que estaban en problemas serios, y que finalmente pudieron ser rescatados.

McConnell (1997) cita a Lakhanpal, estableciendo que los directivos, administradores o líderes de proyectos normalmente escogen a los miembros del proyecto basándose en el nivel de experiencia y en las posibilidades individuales. Así mismo sugiere que los directivos que están preocupados por formar sus equipos de trabajo, deberían de asignar a los programadores basándose en sus habilidades para contribuir primero a formar un equipo unido, y después basarse en sus habilidades individuales.

Debido a este enfoque de selección de personal para el equipo de trabajo, muchos administradores o líderes de proyectos, fallan al seleccionar a su personal, ya que en múltiples ocasiones se puede seleccionar mal al personal debido a razones equivocadas, por ejemplo, se suelen elegir porque tengan un cierto interés en el proyecto, porque cobran poco o, lo más normal, porque simplemente están disponibles. De acuerdo con McConnell (1997), los miembros de un equipo se tienen que elegir comprobando quiénes poseen realmente las aptitudes que se necesitan, siendo las más importantes:

1. Las habilidades técnicas específicas: área de aplicación, plataforma, metodologías y lenguaje de programación
2. Un fuerte deseo de colaborar
3. Habilidades de colaboración específicas que son necesarias para trabajar eficientemente con otros

Dentro de un Depto. de Informática, puede ser que existan varios líderes o administradores de proyectos, todos ellos siendo administrados o supervisados por el Gerente de Informática. Por lo tanto, ante la presencia de un nuevo proyecto, Meredith y Mantel (1989) mencionan que antes de desarrollarlo, un Administrador de Proyectos debe ser elegido, siendo el responsable de implementar y completar el proyecto, comenzando desde la lluvia de ideas, y quien seleccione a ese AP es el Gerente de Informática.

Aunque lo más normal es elegir al AP antes de iniciar el desarrollo de un proyecto, éste también puede ser elegido durante en el transcurso del proyecto, cuando se trata de sustituir a alguien que lo dejó, o bien cuando se considere que es saludable para el desarrollo del proyecto el realizar un cambio de AP.

Para poder elegir a un AP, el Gerente de Informática debe de tener en cuenta algunas características que debe de cumplir el seleccionado, dependiendo de

varios factores como disponibilidad de personal y características del proyecto por mencionar algunos, pero fuera de estos dos puntos, Meredith y Mantel (1989) dicen que el individuo que al cuál se le atribuyan las funciones de un AP debe de cumplir con las siguientes características.

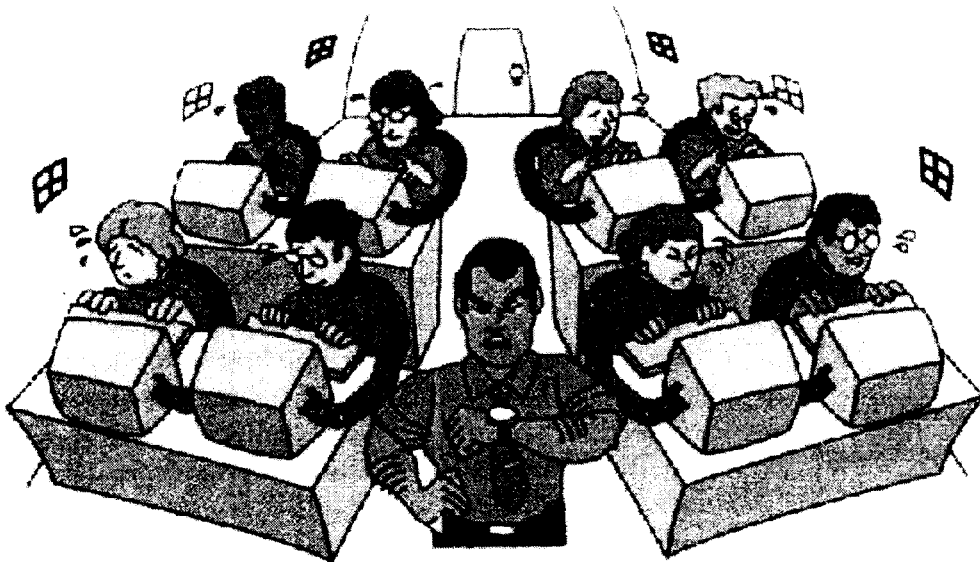
- Un fuerte respaldo de conocimientos técnicos
- Una madurez individual
- Alguien que siempre esté disponible
- Alguien con buenos términos con los ejecutivos
- Una persona que pueda mantener el equipo de trabajo contento
- Alguien que a trabajado en distintos departamentos
- Credibilidad: técnica y administrativa
- Sensibilidad
- Liderazgo

Como podemos apreciar, el AP debe ser una persona que posea además de habilidades técnicas, debe de tener un sentido de liderazgo y participación constante con tanto con el grupo de trabajo para estar al pendiente de los posible conflictos que pudieran presentarse, así como con el cliente para poder establecer una relación agradable de trabajo para que todo marche de acuerdo a lo establecido en los requerimientos.

4.4 El Equipo de Desarrollo

Un equipo de desarrollo es algo más que un conjunto de personas que desean trabajar juntas para formar un equipo. McConnell (1997) cita a Katzenbach y Smith, quienes definen un equipo como "un número de personas pequeño, con habilidades complementarias, que están comprometidas en un propósito, objetivos de rendimiento y enfoque común, en el que todos sean responsables ante todos". Cabe aclarar que no todos los grupos son equipos. Algunos proyectos se pueden llevar a cabo por un grupo de personas que cooperan, pero que no forman un equipo. Lo que hace a un equipo ser exitoso, es la unión que se tenga dentro del mismo, ya que contribuye más a la productividad que las capacidades o experiencia individuales de los miembros del proyecto.

Un equipo de trabajo, no es efectivo porque trabaje bajo presión de tiempo y satisfaga los requerimientos del líder o administrador del proyecto. DeMarco y Lister (1987) mencionan que "la gente que trabaja bajo presión de tiempo, no trabaja mejor, solamente trabaja más rápido". Al trabajar rápido, los miembros del equipo sacrifican la calidad del producto y su propia satisfacción en el trabajo, tal como se ilustra en la figura 4.4.



¿Somos un equipo o no? ¿Por qué vamos tan despacio?
Trabajen más rápido!!

Fig. 4.4 Trabajo en equipo.
[Fuente: McConnell, 1997]

Los equipos productivos a veces se caracterizan por ser equipos que se han ido formando o equipos con gran unión. ¿Qué características tiene un equipo con una gran cohesión, un alto rendimiento y que se ha ido moldeando? De acuerdo con McConnell (1997), el equipo tiene:

- Una alta visión u objetivo compartidos
- Un sentido de identidad del equipo
- Una estructura dirigida por resultados
- Miembros del equipo competentes
- Un compromiso con el equipo
- Confianza mutua
- Interdependencia entre miembros del equipo
- Comunicación efectiva
- Un sentido de autonomía
- Un sentido de enriquecimiento
- Tamaño del equipo pequeño
- Un alto nivel de disfrute

En opinión del mismo McConnell (1997), existen algunos puntos que pueden resultar clave para lograr que un equipo pueda desarrollar sus funciones correctamente y lograr los resultados deseados, así de este modo, será fundamental que los equipos puedan:

- Establecer una misión

- Crear un cambio
- Dirigir al equipo como un equipo
- Delegar las tareas en el equipo, de forma clara, estimulante y con ayuda
- Dejar al equipo los detalles de cómo realizar las tareas
- Tener las tareas bien claras
- Tener un sistema de comunicación efectivo
- Tener un medio de control sobre el rendimiento y retroalimentación
- Las decisiones se deben de tomar, si es posible, basándose en hechos más que en opiniones subjetivas

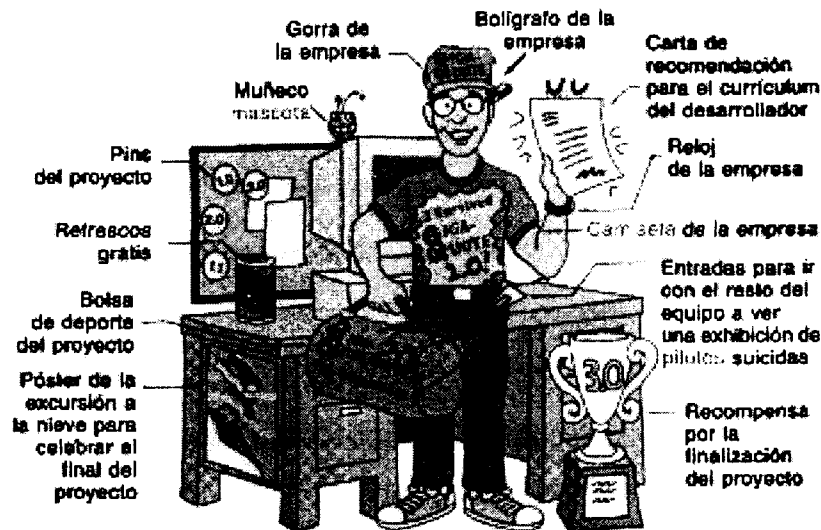


Fig. 4.5 Desarrollador de Software motivado.
[Fuente: McConnell, 1997]

La fig. 4.5 ilustra claramente la identificación de un miembro del equipo de trabajo entregado e identificado tanto con la empresa como con el proyecto en el cuál participa.

Un equipo de trabajo puede ser exitoso, cuando todos los miembros que lo conforman tienen el compromiso e identificación en el grupo, en tanto no lo estén, las cosas no marcharán correctamente. Así como se han mencionado los puntos clave para que un equipo trabaje correctamente, de igual manera McConnell (1997) señala los puntos contrarios, es decir aquellos puntos que originan que un equipo de trabajo no realice sus funciones como se desean, y por consecuencia hay fallas en el trabajo. Las causas de fallo de los equipos son:

- Falta de visión común
- Falta de identidad
- Falta de reconocimiento

- Barreras de la productividad
- Comunicación ineficiente
- Falta de confianza
- Personas problemáticas

Podemos apreciar que si un equipo trabaja armónicamente, y existen buenas relaciones entre sus integrantes, además de estar comprometidos con el proyecto, éste último seguramente será terminado con buenos resultados y podrá cumplir los requerimientos establecidos en la etapa inicial del proyecto. Para poder lograr tener un grupo de trabajo con estas características, Stephen Covey, citado por Weiss y Wysocki (1992), muestra 7 hábitos que sirven como base para administrar a un grupo de trabajo llevándolo a ser efectivo:

1. Ser proactivo
2. Comenzar con el final en la mente
3. Poner las primeras cosas, en primer lugar
4. Pensar Ganar/Ganar
5. Buscar primero entender, para poder ser entendido
6. Buscar sinergias (el equipo da más de sí)
7. Agudizar la vista (buscar el balance)

De igual manera, proporciona una guía que consta de 12 puntos, los cuáles pretenden llevar al AP a aplicar su liderazgo en forma efectiva para poder manejar al grupo de manera adecuada. El AP en opinión de Raudsepp citado por Weiss y Wysocki(1992), deberá de adoptar el estilo de liderazgo que le permita motivar y facultar a los miembros del equipo, así como monitorear y guiar su progreso.

1. Ser directo
2. Reconocer diferencias entre los individuos. Tener una apreciación de las características de cada uno de los miembros del equipo
3. Ayudar a los subordinados viendo los problemas como un cambio u oportunidad
4. Evaluar a los empleados con respecto a su forma de ser creativos, y su contribución
5. Permitir libertad a los miembros de hacer su trabajo como ellos mejor consideren
6. Capacitarse y capacitar a los demás para responder a las propuestas positivas en lugar de reaccionar de manera negativa
7. Desarrollar tolerancia a fallas y errores
8. Proporcionar una atmósfera segura para evitar fallas
9. Ser un recurso humano más que un controlador, un facilitado más que un jefe
10. Actuar como una línea de comunicación entre los empleados y la parte exterior al Depto. de Sistemas

11. Realzar sus habilidades a través de trabajos o seminarios, lecturas especializadas, y ejercicios prácticos
12. Estar seguro de transmitir las ideas innovadoras al jefe con el respaldo y soporte necesario

Se puede apreciar que los puntos anteriormente señalados, indican un claro camino para manejar a la gente, y no tanto a sus cualidades técnicas o a la tecnología con la que se trabaje, mas bien, los puntos tratan de decir que es importante darle su lugar e importancia a cada uno de los miembros del equipo para que éstos se sientan contentos e identificados con el grupo, sabiendo que tienen a un Administrador que no solamente los ve como subordinados sino que también se interesa en su persona y sienta el apoyo del mismo.

Un equipo de trabajo puede trabajar sobre presión de tiempo siempre y cuando se den las condiciones favorables para realizarlo, y que el equipo de desarrollo esté dispuesto a trabajar sobre esta presión, tal es el caso que McConnell (2000b) muestra sobre la empresa Microsoft, la cuál tiene bien entendido que la motivación de su personal es indispensable para un trabajo con calidad y tener gente comprometida con la empresa.

4.5 Relación y responsabilidades de los participantes

Para que un trabajo sea desarrollado en forma adecuada, es indispensable que cada uno de los integrantes, asuma sus responsabilidades en la forma más profesional posible siempre buscando cumplir con cada una de las funciones que le son asignadas. Al seleccionar el equipo de trabajo, de acuerdo con Weiss y Wysocki (1992) el AP debe de considerar varias cualidades de los candidatos a ser incluidos para llevar a cabo el proyecto, pero, entre las más importantes se encuentran:

- Las metas y objetivos del proyecto: tal vez muchos AP solamente buscan a personal con habilidades técnicas, sin considerar al 100% las características y metas a cumplir por el proyecto
- La naturaleza del trabajo técnico a desarrollar
- La experiencia necesaria para reclutar, asignar, delegar, monitorear, comunicar y ejecutar el trabajo requerido en cada parte del proyecto
- Disponibilidad del personal dentro de la organización para desarrollar el proyecto

Al referirnos en primera instancia al AP, Meredith y Mantel(1989) dicen que antes de iniciar cualquier actividad que sea relacionada al proyecto, es recomendable para el AP:

- Conocer por qué existe el proyecto
- Entender la intención del proyecto

- y tener bien definido qué puede llevar al éxito o al fracaso del proyecto

Una vez que el AP logre conocer a fondo y tenga siempre presente los puntos anteriores, entre las primeras tareas a ejecutar como AP, están:

- Realizar un plan de trabajo (calendarización)
- Estimar un presupuesto
- Seleccionar a su equipo de trabajo
- Conocer al cliente
- Asegurarse de que los proveedores estarán disponibles durante el desarrollo del proyecto
- y tener cuidado con los detalles de rutina necesarios para llevar a cabo el proyecto”

El administrador de proyectos llega a tener múltiples responsabilidades, y compromisos tanto con el equipo de desarrollo del proyecto como con la organización y el proyecto mismo, Meredith y Mantel (1989) dicen que fuera de las responsabilidades naturales que nacen con el desarrollo de un proyecto, existen responsabilidades especiales para el AP los cuáles son:

- Adquirir los recursos necesarios y adecuados para desarrollar el proyecto
- Seleccionar y motivar a su personal que conforme el grupo de trabajo del proyecto: uno de los más grandes problemas de todo AP para tener personal con habilidades técnicas, sensibilidad política, orientación a resolver problemas fuertes, fuerte orientación a cumplir las metas, etc.
- Enfrentar los obstáculos
- Definir las metas a lograr
- Negociación

Volviendo la vista hacia las responsabilidades ya no del AP, sino de los miembros del equipo de trabajo, McConnell (1997) nos presenta las siguientes responsabilidades de los equipos de proyectos de software, tomando como referencia a Lakhanpal:

- Desarrollo y revisión de los requerimientos del proyecto
- Desarrollo de la arquitectura del proyecto y de las directrices del diseño que se utilizarán en todo el proyecto
- Definición de aspectos del entorno técnico que se utilizarán en el proyecto (incluyendo lenguajes de programación, compiladores, bibliotecas de código fuente, generadores de código, editores y herramientas de control de la versión)
- Desarrollo de los estándares de codificación que se utilizarán en todo el proyecto
- Coordinar el trabajo en las partes relacionadas del proyecto (incluyendo la definición de interfaces entre subsistemas, módulos y clases)

- Diseño de las partes más complicadas del sistema
- Revisión del diseño y del código de los desarrolladores individuales
- Depurar las partes más complicadas del sistema
- Prueba de requerimientos, diseño y código
- Auditoría del avance del proyecto
- Mantenimiento del software, una vez que se ha construido (incluyendo respuestas a peticiones de mantenimiento y correcciones de emergencia)

Por su parte, Weiss y Wysocki (1992) comentan que no solamente el AP tiene sus responsabilidades. El personal del grupo de trabajo, también tiene sus responsabilidades dentro de la vida del proyecto, el equipo de trabajo exitoso, debe cumplir:

- Compromiso con las metas del proyecto
- Habilidades para comunicarse y compartir la responsabilidad
- Flexibilidad para cambiar de una actividad a otra, dependiendo de la calendarización y necesidades del proyecto
- Competencias técnicas
- Dispuestos a aceptar errores y corregirlos
- Orientado a las tareas
- Habilidades para trabajar dentro de la calendarización y con recursos limitados, o tiempo extra si es necesario
- Habilidad para ayudar, confiar en él y ser ayudado por los demás miembros del equipo
- Ser un miembro del equipo, no un héroe
- Emprendedor pero abierto a sugerencias
- Habilidad para trabajar con 2 o más jefes
- Habilidad para trabajar sin una estructura formal en caso necesario
- Tener el conocimiento y experiencia de herramientas de administración de proyectos

McConnell (1997) presenta en la tabla 4.1 una modificación de Larson y LaFasto, de lo que ellos consideran las directrices básicas para los responsables y miembros de un equipo de trabajo. Si tanto el Líder del proyecto como los demás miembros del equipo desean adoptar un conjunto de reglas, esta tabla es una buena base.

Tabla 4.1 Directrices básicas para miembros y responsables de equipos
[Fuente: McConnell, 1997]

Responsable del Equipo	Miembros del equipo
Como responsable del equipo, tendré que:	Como miembro del equipo, tendré que:
Evitar comprometer los objetivos del equipo con temas políticos	Demostrar una comprensión realista de mi función y posibilidades
Mostrar el compromiso personal con el objetivo del equipo	Demostrar las opiniones sub- jetivas y basadas en hechos
No diluir los esfuerzos del equipo con demasiadas prioridades	Colaborar eficientemente con los demás miembros del equipo
Ser justo e imparcial con todos los miembros del equipo	Poner los objetivos del equipo por encima de cualquier objetivo personal
Estar dispuesto a hacer frente y resolver temas asociados con el rendimiento inadecuado por parte de los miembros del equipo	Mostrar la voluntad de desarrollar el esfuerzo necesario para conseguir el éxito del equipo
Estar abierto a nuevas ideas e información por parte de los miembros del equipo	Estar dispuesto a compartir información, percepción y retroalimentación apropiadamente
	Proporcionar ayuda a los demás miembros del equipo cuando la necesiten
	Demostrar altos niveles de excelencia
	Apoyar y admitir las decisiones del equipo
	Demostrar que se es consecuente con sus principios, enfrentándose a temas importantes
	Demostrar aspectos de responsabilidad de forma que contribuya al éxito del equipo
	Responder constructivamente a la retroalimentación de los demás

El tener gente bien capacitada para realizar sus actividades de manera efectiva es fundamental en cualquier tipo de empresa, y más aún cuando se trata de trabajar en equipos cuyo proyecto es de gran escala, en donde se necesita gente capaz de organizarse de la mejor manera para prevenir que los errores puedan presentarse. Para que esto suceda, es necesario que el personal tenga las habilidades para manejar algunas herramientas como lo menciona Curtis (2000), quien dice que el conocer las herramientas que se pueden usar harán que el trabajo sea mucho más sencillo de realizar.

Concordando con lo expuesto anteriormente, podemos hacer notar que un Administrador de Proyectos es el responsable de organizar, formar su equipo de trabajo, presupuestar, dirigir, planear y controlar el proyecto. En otras palabras, el Administrador de Proyectos lo maneja todo, pero los Administradores Funcionales pueden involucrarse en la selección de la tecnología a utilizar por el proyecto y por usuarios específicos que trabajaran en él.

4.6 Conclusiones

Si cada uno de los participantes, tanto del equipo de trabajo, el Administrador de Proyectos y el cliente conocen sus funciones a la perfección, y están comprometidos con el desarrollo del proyecto, lo más seguro es que finalmente se pueda llegar a formar una buena relación de trabajo entre todos ellos, conllevando con su buen desempeño a lograr las metas propuestas en los requerimientos iniciales del proyecto.

Por el contrario, si existe algún conflicto en alguna de las partes, es necesario que se solucione de la forma más amena y pronta posible para que afecte en lo menos que se pueda al proyecto. El Administrador de Proyectos no es el único responsable de que un proyecto termine bien o mal, todos los participantes tienen un grado de responsabilidad, así que esta filosofía se debe de hacer sentir en cada uno de ellos, para evitar asperezas y retrasos o altos costos en el proyecto que se trabaja.

El anexo 6 muestra los elementos que son necesarios para el desarrollo del proyecto, proporcionando una serie de tareas o procesos así como las habilidades y cualidades personales que deberían de tener cada uno de los participantes. Un factor común en todos, es que debe de existir siempre una excelente comunicación a lo largo del desarrollo del proyecto, esto evitará que se presenten confusiones en requerimientos, duplicidad de funciones, generará un adecuado canal de retroalimentación, entre otros múltiples beneficios. Si todos estos factores se encuentran en estos tres elementos base (usuario, líder del proyecto y equipo de trabajo), se tendrá una buena base para resolver cualquier conflicto que se presente e intente contra la integridad del sistema que se desarrolla.

Capítulo V

Investigación de Campo

5.1 Introducción

El desarrollo de sistemas de información involucra múltiples actividades, que abarcan los puntos económicos, humanos, materiales y tecnológicos. Cada uno de los desarrolladores tiene su propia forma de identificar los problemas a resolver, así como también sus preferencias por alguna plataforma de desarrollo y estructura de trabajo. Así tenemos de esta manera, múltiples caminos para poder desarrollar un nuevo sistema de información. El uso de una metodología de desarrollo depende de varias variables, ya sea a la naturaleza del proyecto, al tiempo que se tenga disponible para llevarlo a cabo, del presupuesto asignado, y la preferencia del desarrollador por utilizar alguna metodología en especial.

Cada desarrollo de un sistema de información nos presentará inconvenientes conforme este se vaya desarrollando, y entre más complejo sea el proyecto, más cuidado se deberá de tener en cuidar algunos puntos administrativos, en asignar adecuadamente las tareas a los miembros del equipo, en general, poner mas cuidado a todas las actividades concernientes a dicho proyecto, con el objetivo de minimizar los errores que modifiquen el plan inicial de trabajo.

5.2 Objetivos de la Investigación.

Los objetivos generales de la investigación de campo son:

- Identificar los factores de asignación de tareas en el desarrollo de un sistema de información.
- Identificar los indicadores más comunes en las empresas de desarrollo de sistemas de la ciudad de Monterrey y área urbana que afectan el desarrollo del proyecto.
- Identificar las posibles causas que originan que los indicadores sigan apareciendo en el desarrollo de un sistema de información.
- Determinar cuáles son las posibles áreas de mejora para que estos indicadores tiendan a desaparecer.
- Identificar si se fomenta el trabajo en equipo, factor primordial en el desarrollo de un proyecto.

- Identificar la metodología de desarrollo más común.

Estos puntos anteriores nos llevarán a verificar qué puntos de los expuestos en los capítulos anteriores, se presentan actualmente en las empresas seleccionadas en Monterrey y en el área metropolitana, además de poder apreciar cuáles son los que mayor presencia tienen, y cuáles están siendo evitadas por los equipos de trabajo. Esto nos permitirá proponer alguna alternativa para realizar trabajos futuros.

5.3 Criterios de selección de empresas.

Para llevar a cabo la investigación de campo, se utilizaron los siguientes criterios para la selección de las empresas:

- Se seleccionaron aquellas empresas cuyo giro principal es el Análisis de Sistemas y Procesamiento Informático ubicadas en el área metropolitana de Monterrey, N.L.
- La encuesta está dirigida a Líderes de Proyectos y/o Gerentes de Informática, con experiencia en el desarrollo de sistemas de información.
- Las empresas seleccionadas deben de estar ubicadas en la pequeña y mediana empresa.

Se seleccionaron únicamente empresas ubicadas en la ciudad de Monterrey y su área metropolitana, debido a que es más fácil tener acceso a los líderes de proyecto que en ellas trabajan, ya que si se contactaran empresas fuera de la ciudad de Monterrey y su área metropolitana, no habría una gran facilidad para desplazarse a dichas empresas para poder entregar la herramienta que utilizada.

5.4 Muestra utilizada.

Para obtener la muestra representativa aceptada estadísticamente se utilizó la siguiente fórmula (Mend, 1981):

$$n = \frac{Nz^2pq}{V^2(N-1) + z^2pq}$$

donde

n=tamaño de la muestra;

z=nivel de confiabilidad, generalmente se usa 95% (0.95);

P=probabilidad de éxito de que las personas que se seleccionen tengan las características que se requieren;

e=error esperado;

N=tamaño de la población

Para obtener la muestra se utilizaron los siguientes valores:

n=?

z=1.645

p=0.5

q=0.5

V=0.10

N=63

$$n=63(1.645)^2(0.5)(0.5) / (0.01)(62) + (1.645)^2(0.5)(0.5)$$

$$n=63(2.706025)(0.25) / (0.01)(62) + (2.706025)(0.25)$$

$$n=42.61989375 / 0.62 + 0.67550625$$

$$n=42.61989375 / 1.29650625$$

$$n=32.87287951755$$

De tal forma que el tamaño de muestra es de 32 empresas, siendo la misma de un 50.79% del total de las empresas proporcionadas por el INEGI (Instituto Nacional de Estadística, Geografía e Informática) utilizando el sistema SAIC (Sistema Automatizado de Información Censal).

5.5 Herramientas y variables de la investigación.

Para lograr obtener lo que los diferentes Líderes de Proyecto y/o Gerentes de Informática de las diferentes empresas seleccionadas en el área metropolitana de Monterrey, se diseñó una encuesta, basada en preguntas cerradas, dando la oportunidad a quien la contesta, de seleccionar aquella(s) opción(es) más adecuada de acuerdo a su experiencia profesional. En algunas preguntas, la respuesta debe de ser única, en algunas otras, la respuesta puede tener varios reactivos. La encuesta se encuentra listada en el Anexo 1.

De acuerdo con Murray mencionado por De León (1997), la prueba estadística utilizada son las tablas frecuentes o distribución de frecuencias las cuales sirven cuando se tiene un gran número de datos que necesitan ser distribuidos en clases y categorías para determinar el número perteneciente a cada categoría.

Por su parte, las variables utilizadas las encontramos en el Anexo 2. Las preguntas en gran parte solicitan un valor de 1 a 5, en donde 1 es el total desacuerdo al punto en cuestión, y 5 representa el total acuerdo, dando la opción NA para cuando el cuestionamiento no sea aplicable de acuerdo al criterio del encuestado. Las opciones 1, 2 y 3 significa que no existe ese parámetro en su organización, las opciones 4 y 5 establecen que dicho parámetro o variable está presente en la organización.

5.6 Conclusiones

Con los objetivos establecidos se podrá determinar qué es lo que pasa actualmente en las empresas en cuanto al desarrollo de sistemas de información. Con ellos se podrá identificar los puntos más importantes y de trascendencia en el ciclo de vida de un sistema de información, tales como los errores en los que se cae frecuentemente, identificar el nivel de comunicación y trabajo en equipo que persiste en las organizaciones, las metodologías de desarrollo más utilizadas pero no por esto indique que sean las mejores, además de identificar cuáles serían las áreas de mejora que se pueden lograr para que no sigan existiendo los errores actuales en el desarrollo de los sistemas de información.

CAPITULO VI

Resultados de la Investigación de Campo

6.1 Introducción

En este capítulo, se muestran los resultados encontrados en la investigación de campo realizada en las empresas seleccionadas del área metropolitana de Monterrey, cuyo giro principal es el Análisis de Sistemas y Procesamiento Informático, consta de los siguientes apartados:

- Criterio de asignación de nuevos proyectos
- Desarrollo de sistemas en el último año
- Participación directa del usuario
- Inicio del proyecto
- Comunicación y retroalimentación
- Metodologías de desarrollo
- Indicadores de falla presentados en el último año en el desarrollo de sistemas de información
- Origen de fallas
- Areas de mejora

Las gráficas mostradas a continuación en este capítulo, son resultado de las encuestas entregadas a los líderes de proyecto de las empresas visitadas con el objetivo de recabar la información y analizar qué es lo que pasa actualmente en la ciudad de Monterrey y área metropolitana en empresas de este sector.

6.2 Criterio de asignación de nuevos proyectos

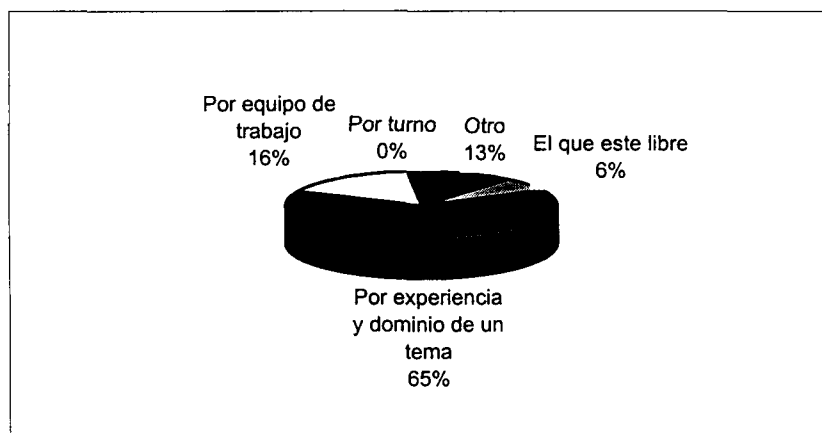


Fig. 6.2 Asignación de nuevos proyectos

De acuerdo a los resultados obtenidos en las encuestas aplicadas, la fig.6.2 muestra que la mayoría de los Líderes de Proyecto asignan los nuevos proyectos, de acuerdo a la experiencia y dominio de algún tema en especial que posea su personal, y ninguno asigna responsabilidades por turno, esto significa que se tiene bien contemplado cuáles son las características del personal con que cuentan y saben de qué es capaz cada uno de sus integrantes.

6.3 Desarrollo de sistemas en el último año

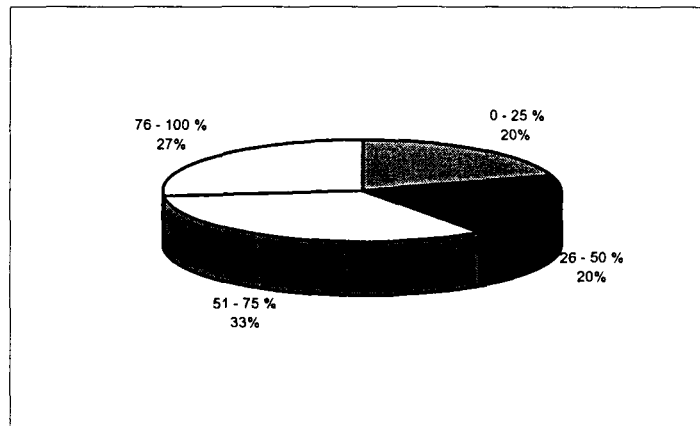


Fig. 6.3.1 Entrega de sistemas en fecha establecida y sin demora alguna

La cantidad de sistemas desarrollados durante el último año, y que fueron entregados sin alguna demora, conforme a lo mostrado por la fig. 6.3.1 es equivalente a poco menos de la tercera parte del total de sistemas (27%), esto indica, que aún, en la mayoría de los casos, siguen presentándose algunos detalles que afectan las fechas programadas para entrega de los sistemas.

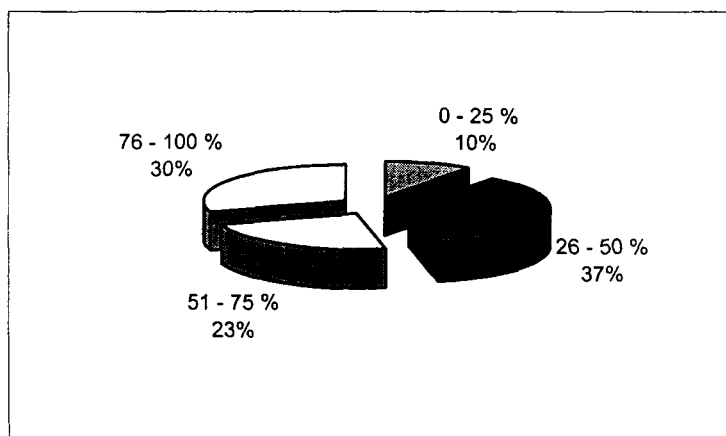


Fig. 6.3.2 Modificaciones a los requerimientos

La fig. 6.3.2 ilustra, que una de las situaciones que pueden ocasionar que los sistemas no se entregan a tiempo en la fecha programada, es precisamente los cambios de requerimientos, ya que el 75% de la totalidad de sistemas desarrollados presentaron modificaciones en los requerimientos una vez que se a comenzado el proyecto. Esto significa que en un proyecto, es común que se presenten cambios en los requerimientos por parte del usuario una vez que se a comenzado a trabajar en el mismo, por lo tanto es necesario que se establezcan procedimientos de control de cambios en los requerimientos del sistema, que busquen de alguna manera minimizar el impacto que estos puedan llegar a tener en el desarrollo del proyecto.

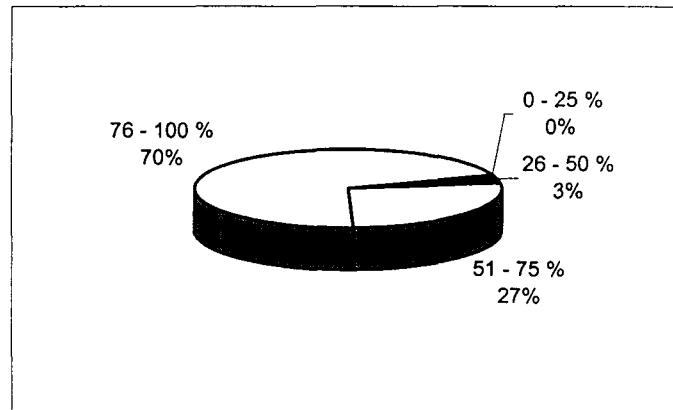


Fig. 6.3.3 Cumplimiento de expectativas del usuario

A pesar de los retrasos en tiempo que muestra la fig. 6.3.1, la entrega aunque sea tardía de los sistemas, como lo muestra la fig. 6.3.3, cumplen satisfactoriamente los requerimientos del usuario en su mayor parte (70%), en tanto un 30% restante no cumple al 100% las necesidades del usuario. Esto indica que el porcentaje que no cumple con las satisfacciones del usuario, o no han sido liberados en su totalidad, o bien, su entrega se vio precipitada por alguna razón, aunque también puede existir la posibilidad de que el análisis de los requerimientos del usuario no haya sido satisfactorio, y en el transcurso del desarrollo, no se realizaron evaluaciones intermedias para verificar que las funciones del sistema eran las que el analista había solicitado.

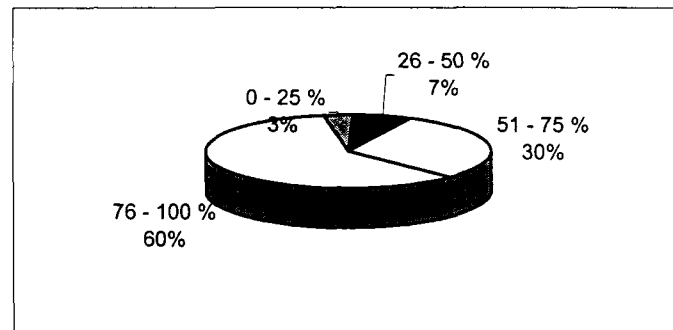


Fig. 6.3.4 Uso adecuado de los sistemas

La fig.6.3.4 muestra que la mayor parte de los sistemas (60% del total), tienen un uso adecuado por parte del usuario, debido a la capacitación que recibió éste último sobre el manejo del sistema, aunque existe un porcentaje realmente alto (40%) que no se usa adecuadamente, debido a la ausencia de una adecuada capacitación sobre su uso, cuestión que pudiera provocar que no se aproveche al 100% las funciones del sistema, y posiblemente debido a este mal uso, el sistema pueda llegar a retirarse.

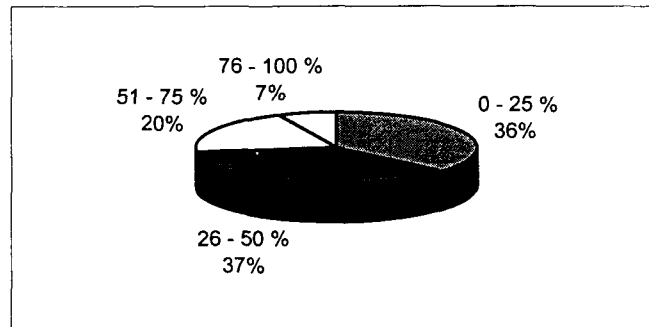


Fig. 6.3.5 Modificación por al menos 1 problema presentado

La fig. 6.3.5 muestra que en la mayoría de los sistemas (64%) se presenta al menos una situación que provoque modificaciones en el calendario de desarrollo del sistema, esto invariablemente repercute directamente en el tiempo estimado de entrega, en el presupuesto, en la calidad del proyecto, entre otras áreas, y que provocan desvío en el camino de desarrollo del mismo.

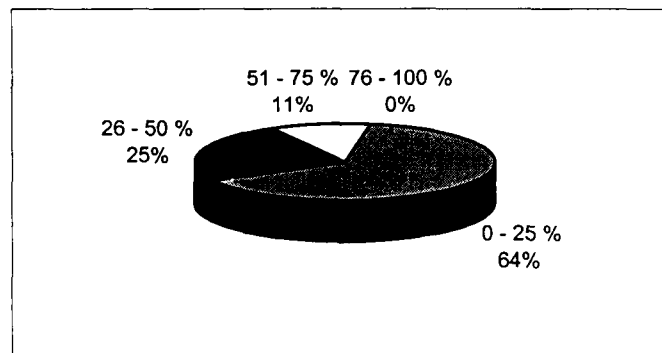


Fig. 6.3.6 Modificaciones por 2 a 5 problemas presentados

La fig. 6.3.6, muestra que el 36% de los sistemas desarrollados, presentaron más de un problema en su desarrollo, esto quiere decir, que en la mayor parte de los sistemas desarrollados, supieron controlar el primer inconveniente y

corrigieron bien el curso del proyecto, no permitiendo que se presentaran más problemas.

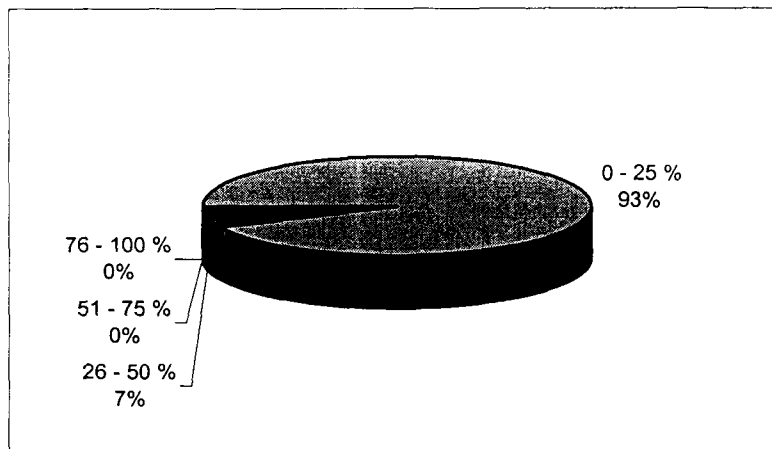


Fig. 6.3.7 Modificaciones por 6 a 10 problemas

Como es de esperarse, de acuerdo al comportamiento ilustrado en la fig. 6.3.7, si no se presentaron de 2 a 5 problemas, mucho menos se esperaría que se presentaran mas de 5 problemas en los proyectos. Sólo el 7% de los sistemas desarrollados, tuvieron más de 5 problemas que los afectaron, un porcentaje bajo, mas no por esto es permisible. Esto indica que existe una carencia de análisis a profundidad, y una carencia total de un adecuado plan de trabajo y administración sobre las actividades que se deben de desarrollar para llevar a cabo el proyecto.

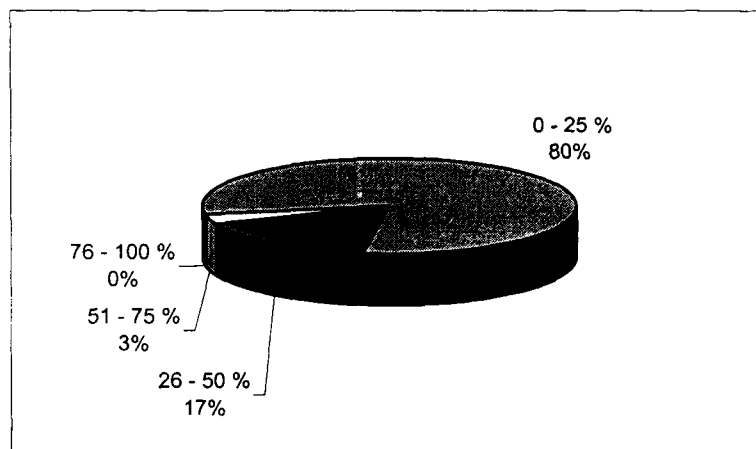


Fig. 6.3.8 Problemas ocasionados por la Tecnología de Información

La fig. 6.3.8 muestra que los problemas que se presentan en el desarrollo de un sistema de información, durante el último año no fueron ocasionados en su

mayoría por el uso inadecuado o falta de tecnología de información adecuada a las necesidades del proyecto. Esto significa que se cuenta por lo general con la tecnología adecuada y acorde al proyecto, además de que el personal tiene las habilidades técnicas para poder utilizarlas adecuadamente.

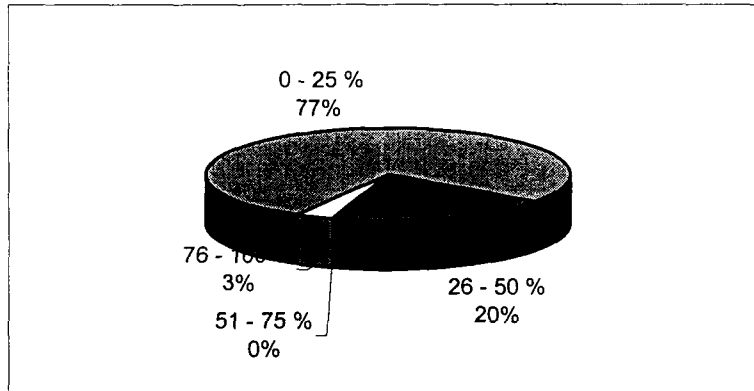


Fig. 6.3.9 Modificaciones por falta de habilidades del equipo de trabajo

Respecto a las habilidades del equipo de trabajo, la fig. 6.3.9 muestra que solamente un 23% de los sistemas, presentaron problemas por este indicador, esto indica, que los equipos de trabajo, en su mayoría, son formados adecuadamente de acuerdo a las necesidades del proyecto.

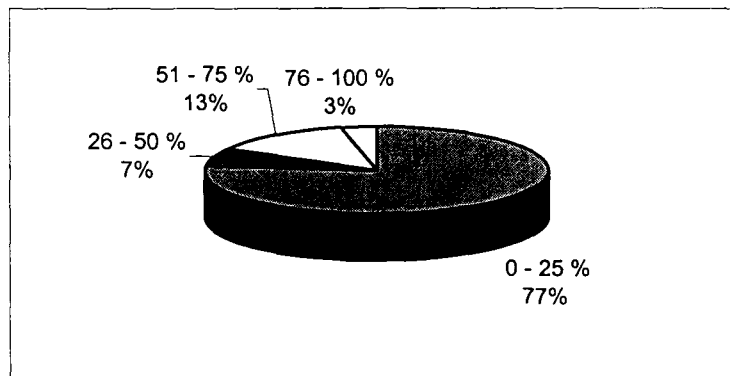


Fig. 6.3.10 Modificaciones presentadas por problemas económicos

La fig.6.3.10 muestra que uno de los factores que afectan a todo tipo de proyecto, como lo es el presupuesto, afectó en un 23% de la totalidad de los sistemas desarrollados, por lo tanto, esto indica que sigue existiendo una mala planeación de recursos requeridos, o bien pudieron haber sido afectados por sucesos inesperados.

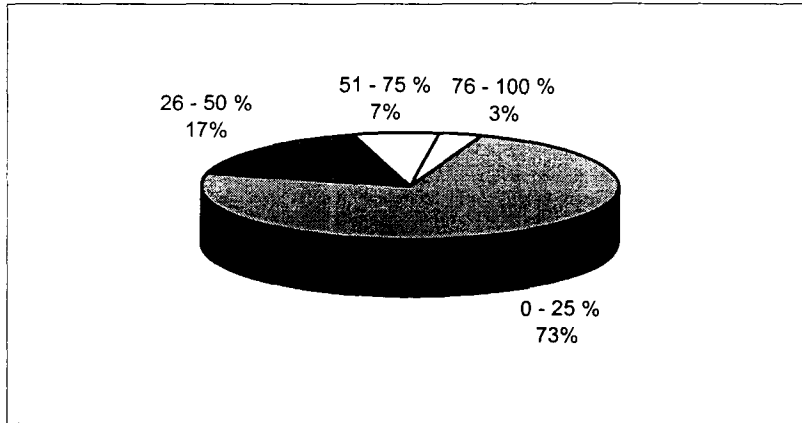


Fig. 6.3.11 Modificaciones presentadas por falta de personal

La fig. 6.3.11 muestra que se presentan problemas por falta de personal en un 27% de los casos, esto quiere significar que no se tiene el tamaño del equipo de trabajo adecuado, puede deberse a falta de presupuesto para contratar gente, o simplemente se calculó erróneamente la cantidad de gente que debería de trabajar en el proyecto.

6.4 Participación directa del usuario

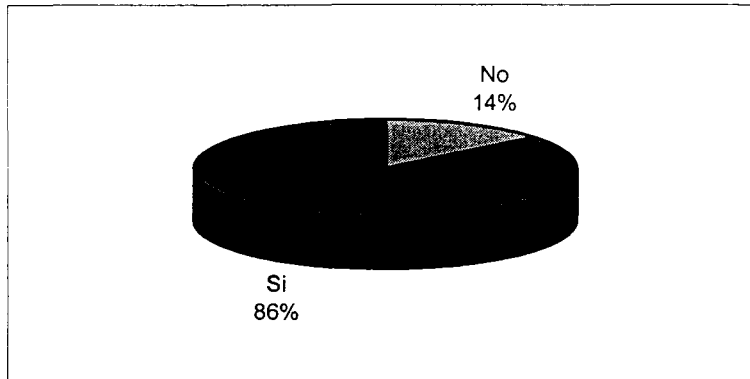


Fig. 6.4.1 Participación del usuario en la Identificación de las necesidades

La fig. 6.4.1 muestra, que en el 86% de los proyectos realizados, el usuario tiene una participación directa con los analistas para determinar exactamente sus necesidades reales, y en un 14%, el usuario no tiene una participación 100% activa en la determinación de los mismos. Esto indica que el usuario que no participa directamente en la identificación de las necesidades, es porque no sabe externar las mismas, o bien, el sistema será impuesto por su jefe inmediato siendo este último quien decide qué es lo que deberá de realizar el sistema.

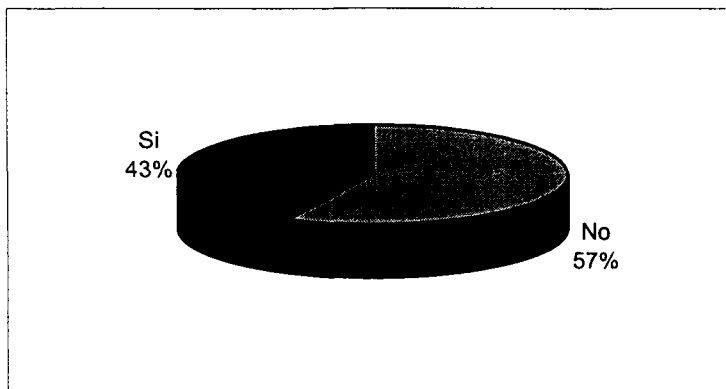


Fig. 6.4.2 Participación del usuario en el estudio de factibilidad

La fig.6.4.2 muestra que el usuario, en la mayoría de los proyectos, no es tomado en cuenta en el estudio de factibilidad de desarrollo del proyecto, esto significa que el usuario no decide si se puede o no realizar, el estudio es realizado por el líder del proyecto y autorizado por sus jefes inmediatos, ya que hay que determinar varios factores como el económico, legal y financiero además del operativo.

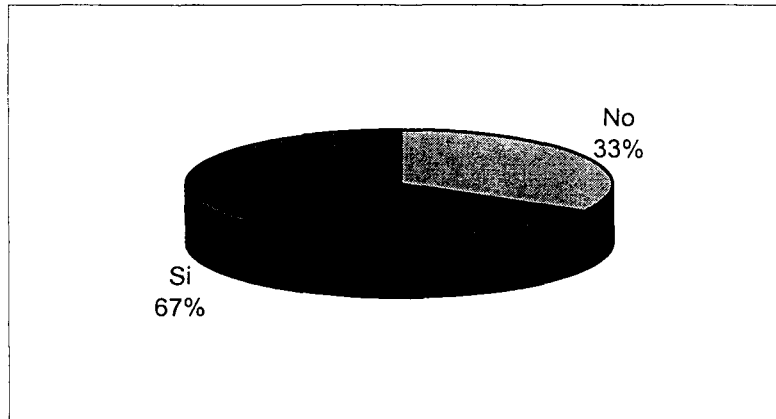


Fig. 6.4.3 Participación del usuario en el análisis

La fig. 6.4.3 muestra que el usuario final del sistema, tiene una participación bastante activa en el análisis del proyecto, esto indica que el trabajo es en conjunto con el equipo de trabajo, particularmente con los analistas, haciendo más claro lo que el mismo sistema debe de cumplir.

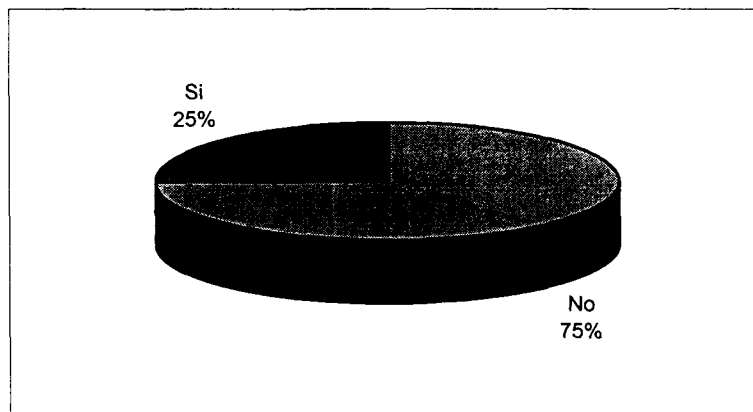


Fig. 6.4.4 Participación del usuario en el diseño lógico

Para la etapa del diseño lógico, conforme a lo mostrado por la fig.6.4.4 la participación del usuario disminuye, siendo esta función realizada por los especialistas del equipo de trabajo, aunque en un 25% de los sistemas desarrollados, aún toman en cuenta la opinión del usuario. Esto significa que el usuario, solamente debe de participar a lo más en la determinación de los requerimientos, en determinar cuáles son sus necesidades, y no debe de ser tomado en cuenta para una etapa posterior.

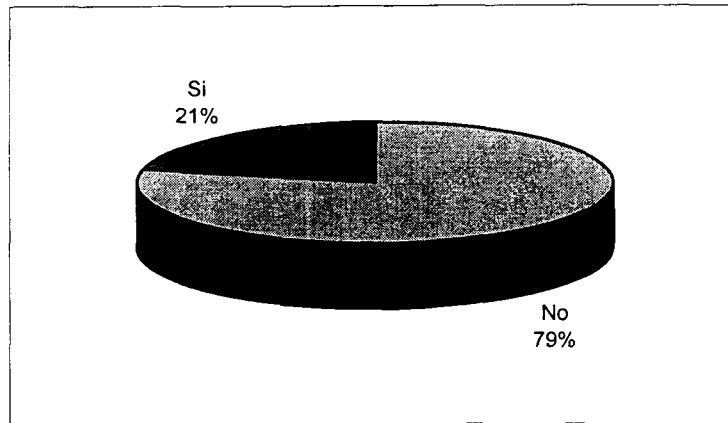


Fig. 6.4.5 Participación del usuario en el diseño físico

En la etapa del diseño físico, como se muestra en la fig. 6.4.5, sigue presente la participación del usuario, aunque se puede observar que su participación disminuye en comparación con el diseño lógico. Esto indica que algunas empresas consideran valiosas las posibles contribuciones que el usuario pudiera aportar a los diseñadores, ayudándose mutuamente esclareciendo las funciones que realizará el sistema.

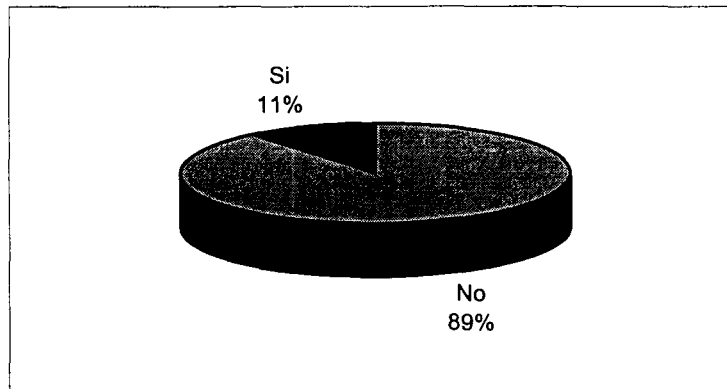


Fig. 6.4.6 Participación del usuario en la codificación

Pareciera ilógico que un usuario participe dentro de la fase de codificación, ya que ésta debería de ser una función exclusiva de los programadores, mas sin embargo la fig.6.4.6 muestra que existe la opción de que el usuario sea el mismo programador, esto es para aplicaciones personales, aunque puede observarse que la participación no es muy elevada (11% del total de sistemas desarrollados).

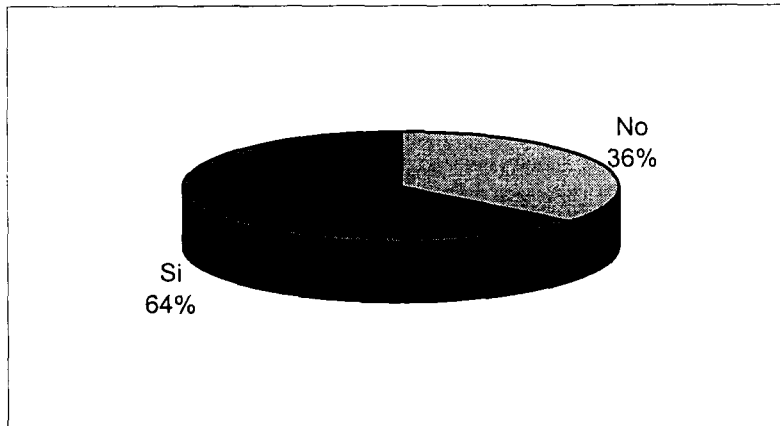


Fig. 6.4.7 Participación del usuario en la etapa de pruebas

En la etapa de pruebas, la fig. 6.4.7 ilustra que la participación del usuario es importante, mas sin embargo no es total, el usuario es el indicado para probar si las funciones que realiza el sistema son las adecuadas a las necesidades, además de que no está familiarizado con el código, como lo estarían los programadores mismos, que prefieren en un 36% no incluir al usuario. Una razón puede ser que el usuario es un cliente externo a la empresa y no tenga la disponibilidad para poder realizar las pruebas y decida dejar en manos del equipo de trabajo las pruebas mismas del sistema.

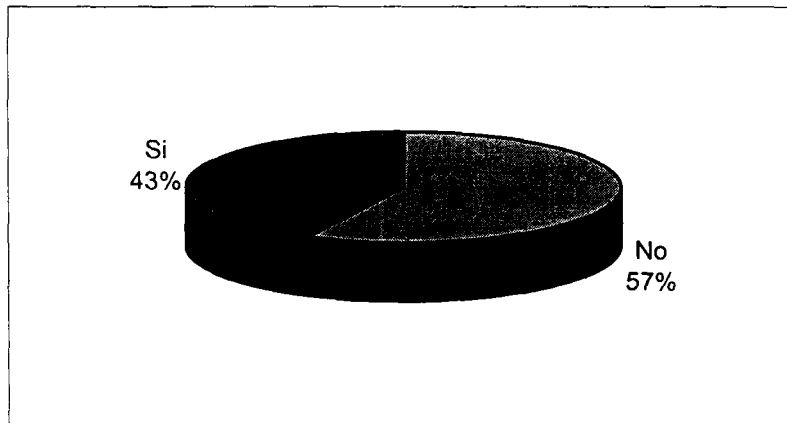


Fig. 6.4.8 Participación del usuario en la etapa de instalación

La participación del usuario disminuye nuevamente en la etapa de instalación del sistema tal como lo muestra la fig. 6.4.8, en donde solamente en el 47% de las ocasiones, el usuario participa en la instalación. Esto indica, que el proceso de instalación del sistema, es una función que deben de realizar en conjunto tanto el personal de informática como el usuario, para que éste este familiarizado con lo más que se pueda referente al sistema, y sepa cómo se debe de configurar el sistema para que trabaje correctamente.

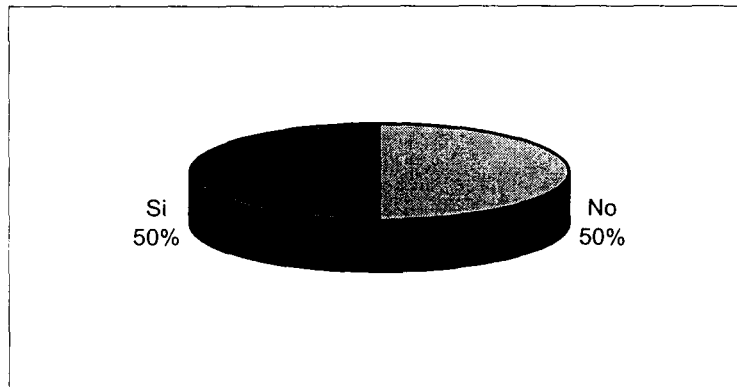


Fig. 6.4.9 Participación del usuario en la etapa de mantenimiento

Por lo que corresponde a la etapa de mantenimiento, la fig. 6.4.9 muestra un porcentaje idéntico tanto para la participación como para la falta de participación del usuario en dicha etapa. El usuario podría participar dentro del mantenimiento, en conjunto con el equipo encargado de proporcionar dicho servicio, para que sepa exactamente que es lo que se hace con el sistema y la integridad de sus bases de datos, además como posible precaución para que en algún momento determinado él pudiera realizarlo directamente.

6.5 Inicio del proyecto

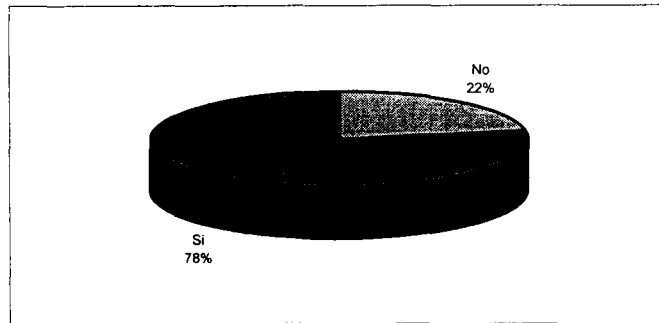


Fig. 6.5.1 Uso de metodologías

La mayoría de los sistemas desarrollados (78% del total), o en este caso, de los equipos de trabajo indican, de acuerdo a la fig. 6.5.1 que sí utilizan una metodología de desarrollo, en tanto que un 22% no utilizan una metodología. Esto indica que posiblemente las actividades se van definiendo de acuerdo a como va trabajando el proyecto, cuestión que si no se tiene una metodología de trabajo, difícilmente el proyecto estará de acuerdo a lo planeado.

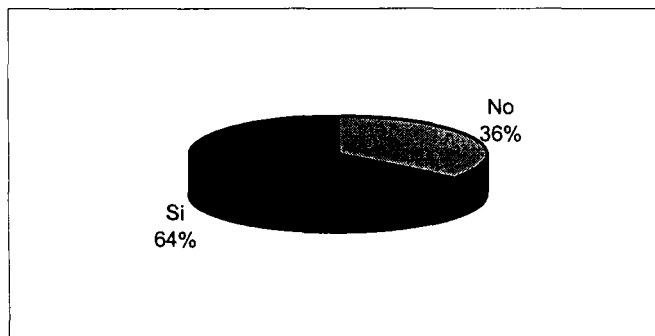


Fig. 6.5.2 Definición de actividades

La fig. 6.5.2 muestra que las actividades a desarrollar dentro del proyecto, son definidas por el líder del mismo, una vez que se ha determinado la metodología a seguir en un 64% de los casos. Esto significa que en el resto de los casos, se definen las actividades antes de seleccionar la metodología, esto puede ser debido a que se tiene una metodología estándar para todos los proyectos (cosa no muy práctica ya que cada proyecto es diferente) o bien se trabaja en conjunto con el cliente previo a la selección de la metodología.

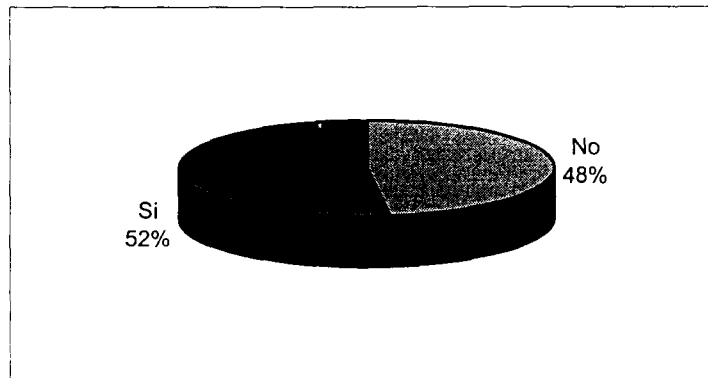


Fig. 6.5.3 Selección y evaluación de la metodología

El líder del proyecto es en el 52% de los casos, conforme a lo mostrado por la fig. 6.5.3, el único responsable de seleccionar y evaluar la metodología a seguir de acuerdo a las características del proyecto. Esto quiere decir que en el resto de los casos se tiene la posibilidad de que se tenga definida una metodología de trabajo propia de la empresa, y no exista la flexibilidad de ser modificada, o bien trabajar de acuerdo a como el usuario o cliente lo desea.

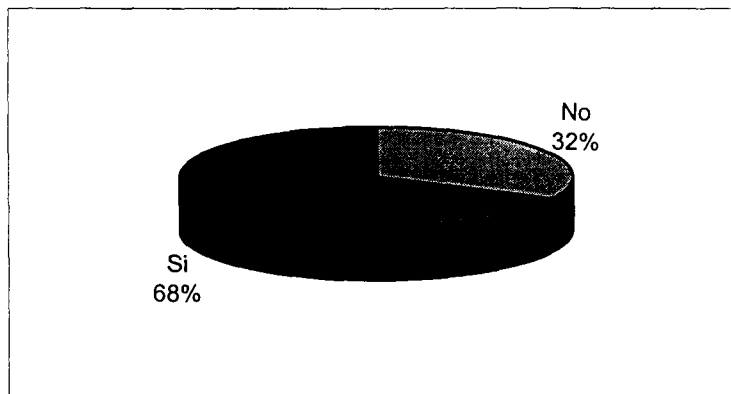


Fig. 6.5.4 Conocimiento de las fases de la metodología

Al trabajar ya en sí en lo que es el proyecto, no en todas las ocasiones los equipos de trabajo conocen a fondo las actividades a desarrollar, únicamente en el 68% de los casos de acuerdo a lo mostrado por la fig. 6.5.4, los integrantes del equipo de trabajo conocen exactamente todas las actividades a realizar, el resto de los casos se muestran paulatinamente de acuerdo a la etapa en la cuál se encuentra el proyecto. Esto significa que de acuerdo a las características de los proyectos y de la metodología de desarrollo seleccionada para trabajar, es como se van a dar a conocer las actividades a los elementos del equipo de trabajo, por ejemplo, en la metodología de prototipo evolutivo, antes debe llevar

a cabo el proyecto en forma definitiva, solamente se entrega un prototipo que ilustrará lo que al término del proyecto pudiera realizar el sistema, esperando evaluar primero dicho prototipo, entonces no es necesario definir todas las actividades cuando aún no se determina si el prototipo muestra exactamente todas las funciones que deberá desarrollar el sistema.

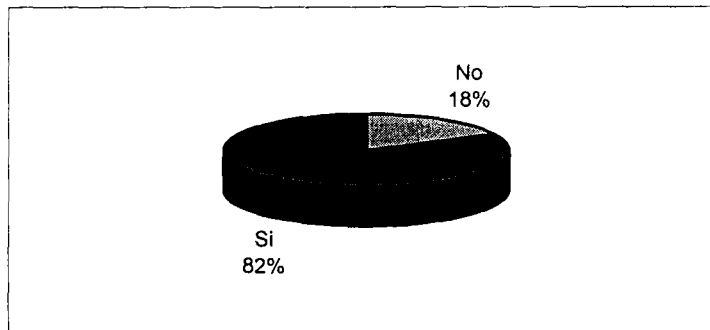


Fig. 6.5.5 Herramientas necesarias

La fig. 6.5.5 muestra que en el 18% de los casos, los integrantes del equipo de trabajo no cuentan con todos los recursos o herramientas necesarias para llevar a cabo sus funciones. De entrada esto es un problema para el equipo, y para la planeación del proyecto, ya que si no se tienen las herramientas necesarias, habrá que ver la manera de conseguirlas, esto implica tiempo y dinero extra en caso de que no se tengan contempladas en la etapa de la identificación de los requerimientos del usuario.

6.6 Comunicación y retroalimentación

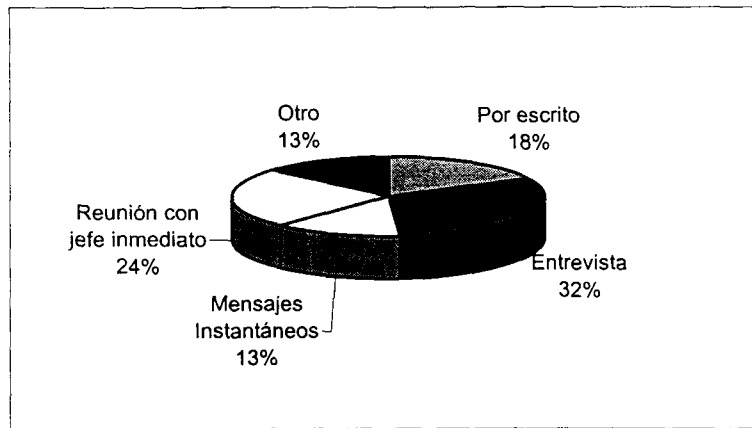


Fig. 6.6.1 Inicio de requerimientos

La fig. 6.6.1 muestra que la forma más comúnmente utilizada por los usuarios para decirle al Director de Informática sus necesidades de desarrollo, es mediante una entrevista en conjunto con su jefe inmediato, el Director de Informática y el mismo usuario. Esto indica que el proceso a llevarse a cabo para solicitar el desarrollo de una aplicación, debe de hacerse en conjunto y de manera formal tanto con el jefe inmediato como con el Director de Informática. Cabe notar que en el 13% de los casos, se utiliza únicamente como etapa inicial el enviar un mensaje por correo electrónico o cualquiera de las aplicaciones de mensajes instantáneos que existen en la actualidad (ICQ, Yahoo Messenger, AOL, etc.), y posteriormente se realizan las juntas necesarias para realizar la identificación de sus requerimientos.

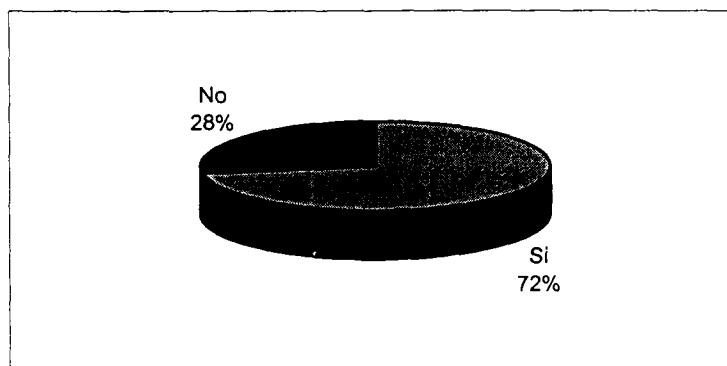


Fig. 6.6.2 Misión del departamento de informática

La fig. 6.6.2 muestra que en el Departamento de Informática, en la mayoría de los casos (72% del total), tienen una misión que tiene una relación directa

con las metas de la organización, en tanto que el 28% no tienen definida alguna misión en particular, sino que la misión de ellos es la misma que la organización tiene en el ámbito corporativo. Esto puede provocar que los elementos que conforman dicho departamento no se sientan identificados con la empresa, con sus valores, provocando de esta manera que pueda presentarse la situación de no comprometerse a fondo con los proyectos de desarrollo y con la empresa misma.

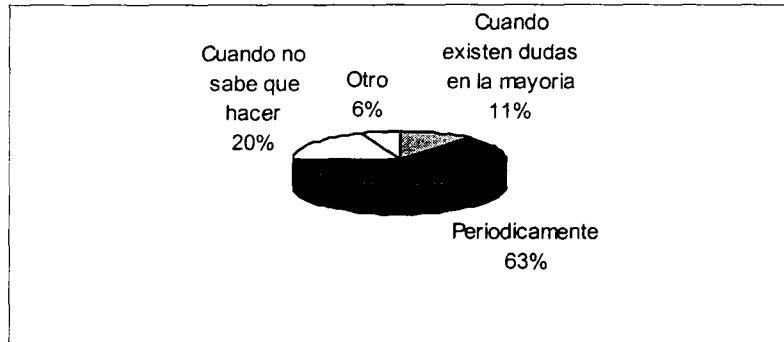


Fig. 6.6.3 Retroalimentación al equipo de trabajo

Una de las funciones básicas e importantes en el funcionamiento de cualquier equipo de trabajo con respecto al desarrollo de algún proyecto es la retroalimentación entre los integrantes del equipo y su líder de proyecto. En este sentido la fig. 6.6.3 ilustra que en el 63% del total de los equipos sí tienen una retroalimentación en forma periódica, esto es cada cierto periodo de tiempo. Esto garantiza un adecuado seguimiento de las actividades de cada elemento del equipo de trabajo, garantizando que las dudas o problemas que puedan surgir, se corrigen a la menor brevedad posible, sin dejar pasar un tiempo largo que vaya a agrandar el problema.

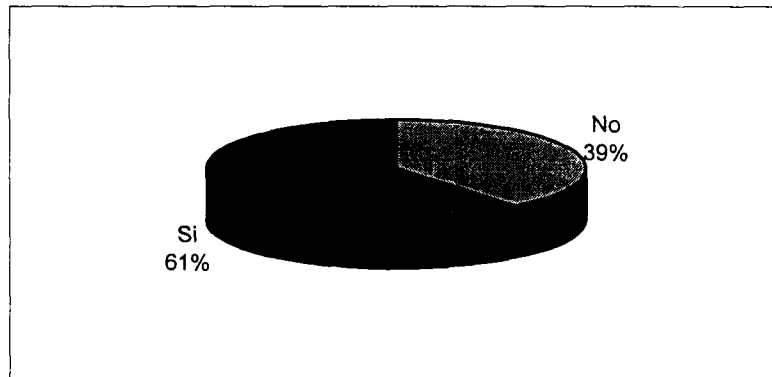


Fig. 6.6.4 Decisiones de la alta gerencia

La comunicación efectiva y directa es importante en cualquier trabajo colaborativo, y en el caso de los proyectos de desarrollo no es la excepción. La fig. 6.6.4 muestra que en el 61% de los casos de desarrollo se cuenta con dicha

comunicación efectiva y a tiempo, en tanto que el 39% no cuenta con una adecuada comunicación. Esto puede traer consecuencias de malos entendidos en el desarrollo de las actividades del proyecto, o bien, que las mismas actividades no sean desarrolladas en el tiempo adecuado, o bien no tengan la misma visión de lo que deba de realizar.

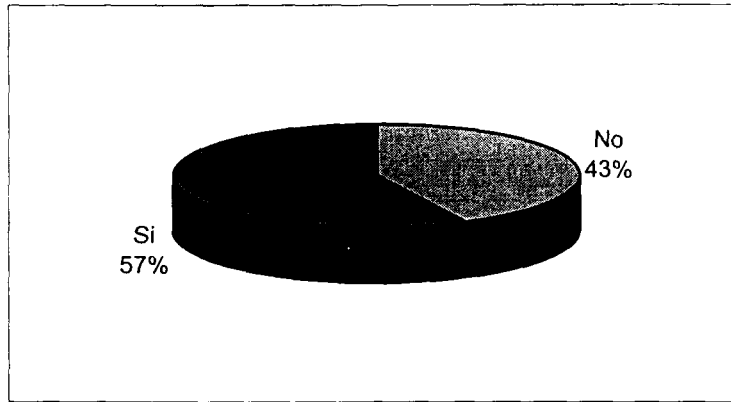


Fig. 6.6.5 Canales bien definidos

Así como la comunicación debe ser directa y en forma oportuna, para que esta se presente, deben existir canales de comunicación bien definidos entre los diferentes departamentos de la organización, o bien entre el usuario o cliente, y el equipo de desarrollo. La fig. 6.6.5 muestra que en el 57% de los sistemas desarrollados, si se tienen los canales bien definidos. El no contar con los mismos, provocará confusiones en el equipo de trabajo, y las actividades no se llevarán a cabo de acuerdo a lo planeado inicialmente.

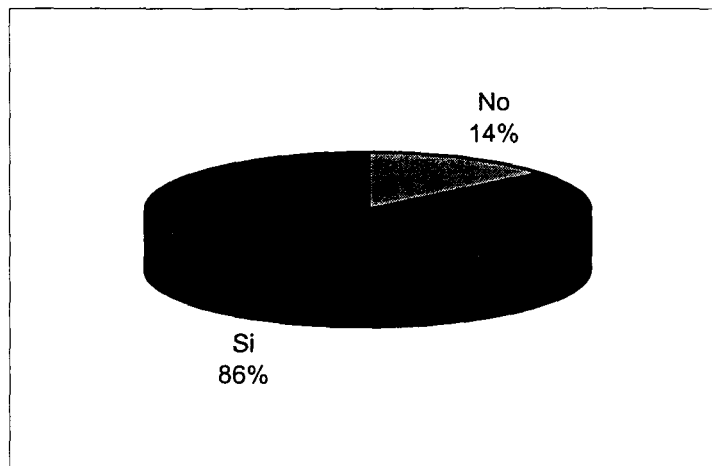


Fig. 6.6.6 Disponibilidad

La fig. 6.6.6 muestra que en el 86% de las ocasiones, el equipo de trabajo muestra una amplia disponibilidad para trabajar, y solamente el 14% no muestra una disponibilidad de hacer sus labores. Esto último indica que los elementos del equipo de trabajo necesitan tener un compromiso mayor con el proyecto, tener sus cargas de trabajo bien distribuidas y no sobrecargadas, o pueden tener una falta de motivación hacia el trabajo.

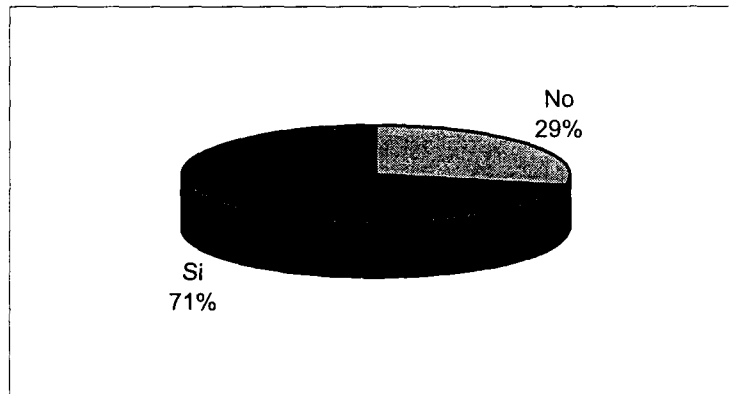


Fig. 6.6.7 Revisión antes de liberación

Todo trabajo en equipo requiere que el trabajo sea supervisado por el líder del proyecto, para poder realizar una liberación en tiempo y condiciones adecuadas (calidad, funcionamiento, capacitación adecuada, etc.). En la fig. 6.6.7 puede verse que en el 71% de los proyectos, el equipo de trabajo requería que su líder de proyecto diera el visto bueno de la liberación del mismo. Esto representa que tanto el equipo de trabajo como su líder, realizar una evaluación de rendimiento y calidad del proyecto, para determinar si se encuentra en condiciones de ser liberado y entregado al usuario para que éste lo pueda utilizar correctamente. En el 29% de los casos, este proceso no es requerido, ya que lo más seguro, es que en el desarrollo del proyecto se incluye una etapa de pruebas que son realizadas por algunos miembros del equipo de trabajo, y pueden ser realizadas en compañía del usuario, sin tener la necesidad de que el líder del proyecto participe en dicha evaluación.

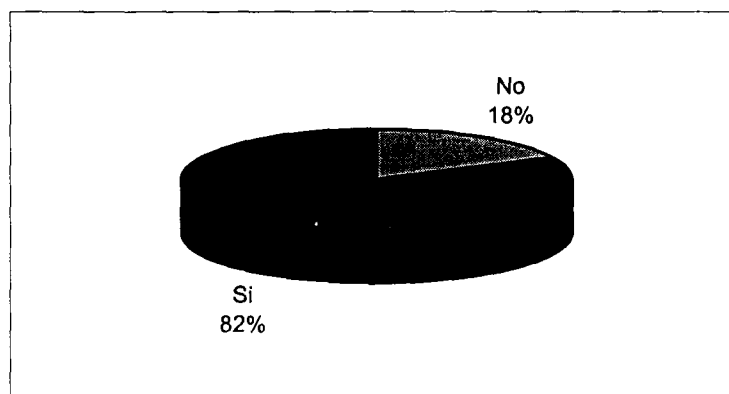


Fig. 6.6.8 Compatibilidad

La fig. 6.6.8 muestra que en el 82% de los casos, al término del proyecto, se evalúa si su funcionamiento está acorde con las metas establecidas, esto es comprobar la compatibilidad de las funciones del sistema de información con las metas previamente fijadas, esta etapa va de la mano con la etapa de pruebas del sistema, y puede ser desempeñada en conjunto con el usuario. Cuando no se evalúa la compatibilidad del proyecto, es debido a que no se cuenta con las condiciones de tiempo necesarias para poder realizarla, teniendo que realizar la liberación sin probar al 100% su compatibilidad.

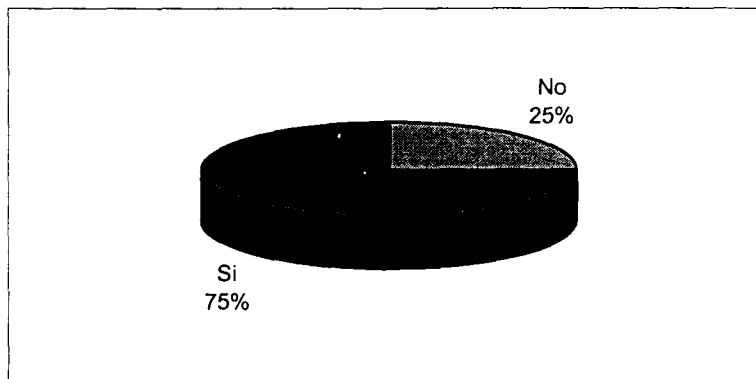


Fig. 6.6.9 Participación al término del trabajo

La fig. 6.6.9 muestra que en el 75% de los casos, los elementos del equipo de trabajo, al terminar sus tareas asignadas del proyecto, colaboran con otros elementos para poder cumplir satisfactoriamente en calidad y tiempo con el mismo, esto corrobora lo mostrado por la fig. 6.6.6 en donde muestra que en la mayor parte de los casos, los miembros del equipo de trabajo muestran una amplia disponibilidad para trabajar. Esto indica que existe un equipo altamente colaborativo, comprometidos con los proyectos que realizan, existiendo un ambiente de trabajo agradable, contando con el apoyo y la colaboración de todos los elementos que conforman dicho equipo.

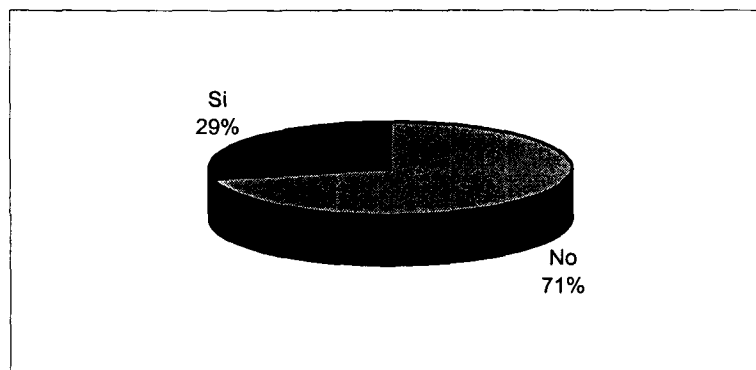


Fig. 6.6.10 Resolución por sí mismo

El 71% de los casos de acuerdo a la fig. 6.6.10 muestra que los integrantes del equipo de trabajo requieren únicamente de su capacidad y la confianza que se deposita en ellos para poder resolver por sí solos aquellos inconvenientes que les impida el desempeñar sus labores de forma correcta. Esto indica que los equipos de trabajo están formados por elementos proactivos, capaces de buscar soluciones sin la necesidad de tener un respaldo por parte de su líder de proyecto.

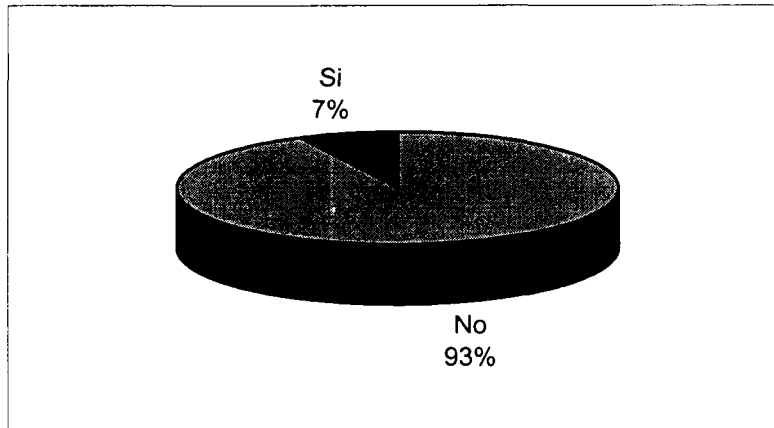


Fig. 6.6.11 Administración por parte del líder del proyecto

La fig. 6.6.11 muestra que en el 93% de los casos, es decir, casi en su totalidad, el líder del proyecto no tiene únicamente entre sus responsabilidades el administrar el proyecto, sino que además participa directamente en las fases de desarrollo, ya sea en el análisis, el diseño, la codificación, instalación, mantenimiento, etc..

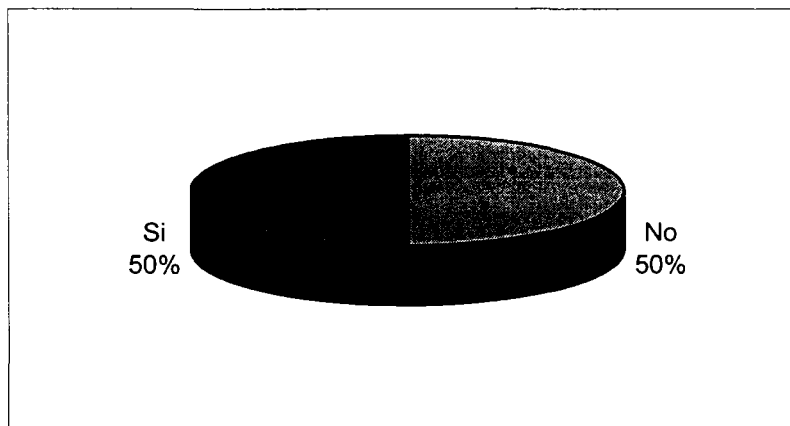


Fig. 6.6.12 Metodología a seguir definida por el líder del proyecto

Contrario a lo que pudiera pensarse, en el que el líder del proyecto es quien deba decidir la metodología a seguir, la fig. 6.6.12 muestra que únicamente en el

50% de los casos se presenta esta situación, esto indica que el trabajo, desde la selección de la metodología, puede ser un trabajo colaborativo entre el líder del proyecto y su equipo de desarrollo, o bien, se toma alguna metodología establecida por la empresa la cuál no sea flexible para modificarse, o bien, se sigue alguna metodología de trabajo que el usuario externo (en caso de ser un cliente externo una empresa) ya tenga definida y únicamente se una el equipo de trabajo al equipo de trabajo de este último.

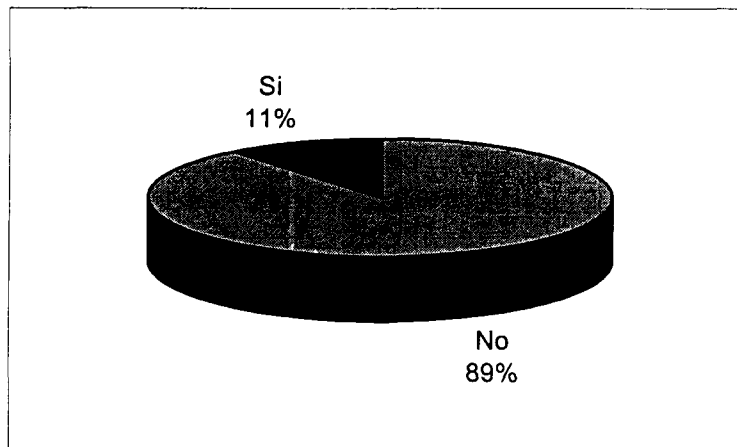


Fig. 6.6.13 Definición de plataforma

La fig. 6.6.13 muestra que la selección de la plataforma de trabajo corresponde en su mayoría al líder del proyecto (89% del total), en tanto que los elementos del equipo de trabajo solamente deciden en un 11% del total. Esto significa que en la mayor parte de los proyectos, los elementos del equipo de trabajo no pueden decidir si un proyecto se desarrollará bajo cierto software o hardware, sino que esto se determina en la etapa de análisis, y es responsabilidad del líder del proyecto el decidir la plataforma.

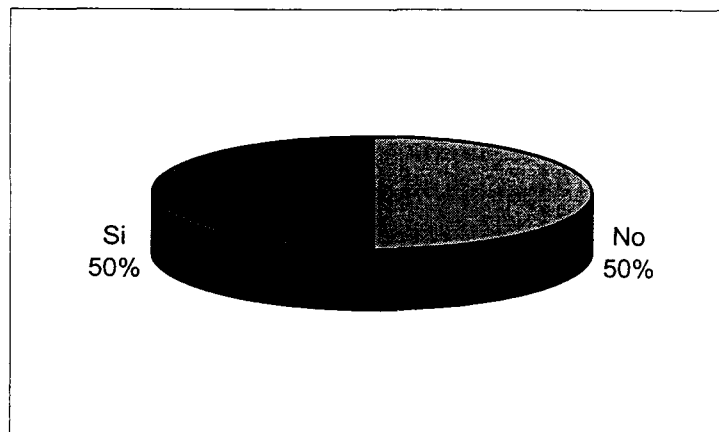


Fig. 6.6.14 Autorización del presupuesto

El presupuesto es uno de los factores más importantes dentro de cualquier tipo de proyecto. En este rubro, la fig. 6.6.14 muestra que el 50% de los líderes de proyecto son quienes autorizan los presupuestos para llevar a cabo algún nuevo proyecto. Esto se presta cuando la empresa desarrolla sus sistemas propios, mas no cuando se desarrolla para un usuario externo (cliente externo), quien es al final de cuentas, quien debe decidir cuanto invertirá en el proyecto.

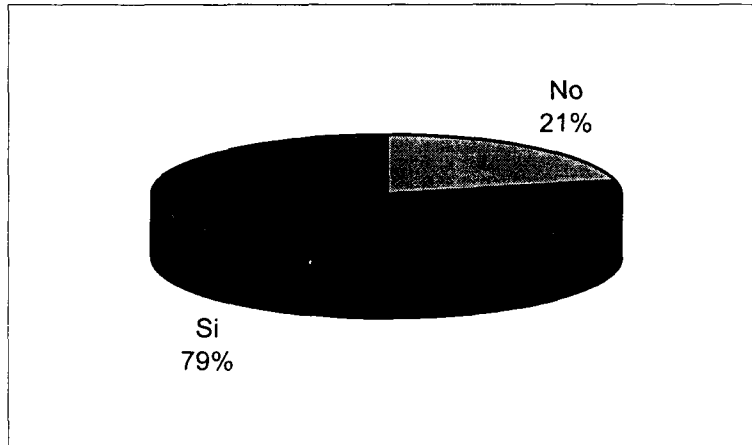


Fig. 6.6.15 Usuario al término del sistema

Es ideal, que al término de cualquier proyecto, y antes de ser implantado para ser utilizado por el usuario, este último emplee parte de su tiempo en capacitarse en el manejo del sistema, para poder tener un adecuado uso del mismo. La fig. 6.6.15 muestra esta situación la cual únicamente se presentó en el 79% de los proyectos, y en un 21% el usuario no ocupa parte de su tiempo laboral en aprender el uso del sistema. Esto último puede llevar a que el uso y explotación del sistema de información que acaba de ser liberado no sea utilizado en forma adecuada, pudiendo obtener resultados erróneos en los datos provocados por su mal uso.

6.7 Metodologías de desarrollo

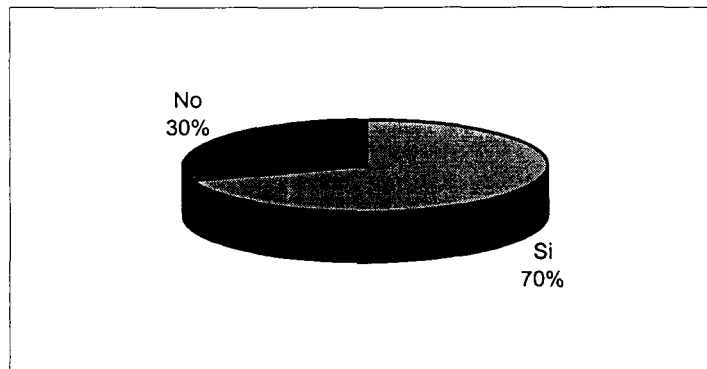


Fig.6.7.1 Metodología favorita

La fig. 6.7.1 muestra que la mayoría de los líderes de proyecto tienen una metodología de desarrollo favorita para utilizarla en el desarrollo de los sistemas de información, en tanto que el 30% no tiene una metodología favorita. Esto indica que utilizan la metodología que más les convenga de acuerdo a las características del proyecto. Esto último sería lo ideal, ya que no todos los sistemas son iguales y algunos requieren de actividades que otros no requieren.

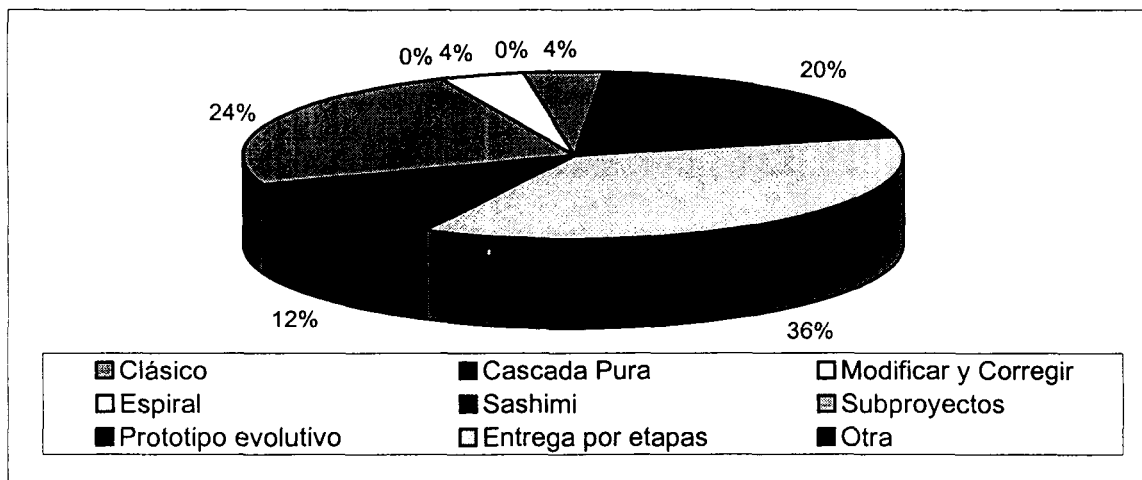


Fig. 6.7.2 Metodología favorita seleccionada

La fig. 6.7.2 muestra las tendencias de favoritismo entre los líderes de proyecto por utilizar alguna metodología en sus equipos de trabajo. Esta gráfica muestra que la metodología que más utilizan es la Entrega por Etapas, en tanto que las de Modificar y Corregir, y la de Sashimi no se utiliza en ningún caso. Esto no quiere decir que dicha metodología sea la mejor para desarrollar

sistemas de información, ni que las metodologías que no tuvieron una sola mención sean las peores. Esto únicamente muestra la preferencia de los líderes de proyectos por alguna metodología en especial.

6.8 Indicadores de falla presentados en el último año en el desarrollo de sistemas de información

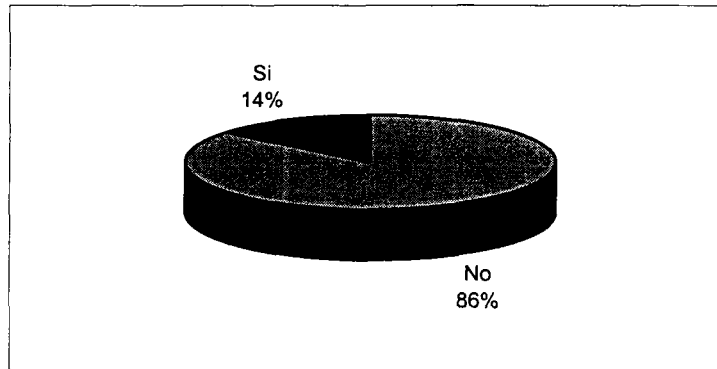


Fig. 6.8.1 No uso de metodología

La fig. 6.8.1 muestra que en el 86% de los proyectos desarrollados, no aplica este criterio, es decir, en el 86% de estos proyectos si se utilizaron metodologías de desarrollo determinadas, y no se llevaron a cabo sin tener un plan de trabajo. En este último punto, únicamente cabe el 14% de los proyectos que se desarrollaron sin seguir una metodología de desarrollo. El seguir el desarrollo de un sistema de información de esta forma, no garantiza que el proyecto vaya a terminar exitosamente, puesto que no se tiene un plan de trabajo, y pudiera ser que el proyecto siempre estuviera en desarrollo, ya que daría la posibilidad de que siempre se hicieran modificaciones, ya que se carece de un plan de trabajo.

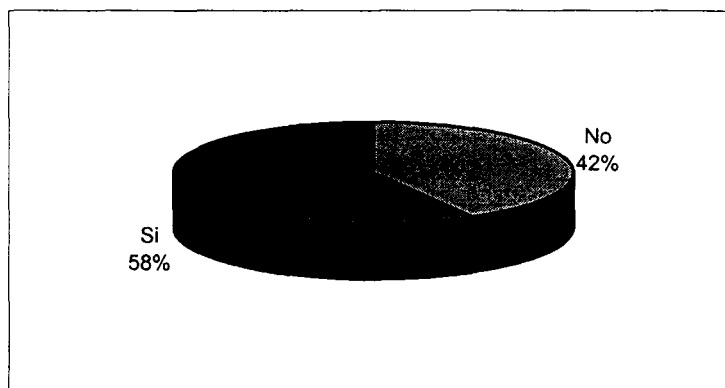


Fig. 6.8.2 Cambios en los requerimientos

La fig. 6.8.2 muestra que en el 58% de los proyectos, el usuario solicita cambios a sus requerimientos iniciales, ocasionando con esto que el proyecto tenga que modificarse. El 42% de los proyectos, no sufrió alteraciones por realizar nuevos requerimientos, esto se deriva de haber realizado un adecuado análisis de los requerimientos y necesidades iniciales del usuario.

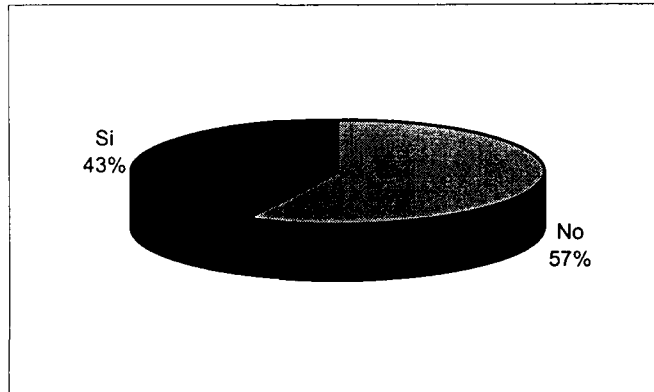


Fig. 6.8.3 Fechas de inicio y/o entrega

Una de las causas de los cambios en los requerimientos ilustrados en la fig. 6.8.2 es el desfase de los tiempos de desarrollo y entrega de los proyectos, tal como lo muestra la fig. 6.8.3, la cuál ilustra que en el 57% de los proyectos si hubo una modificación en este indicador. Esto indica que en un gran porcentaje de los proyectos, se vieron afectados por alguna situación que no se pudo controlar a tiempo, lo cuál originó que dicho proyecto se haya entregado fuera de tiempo.

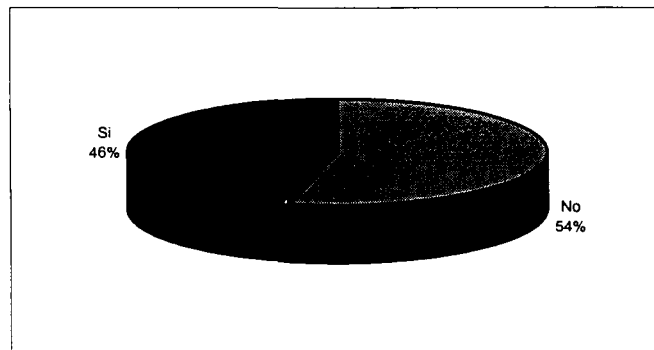


Fig. 6.8.4 Necesidades del usuario

En múltiples ocasiones, el usuario tiene necesidad de un sistema para realizar sus labores, pero no sabe expresar al analista lo que realmente necesita. En ocasiones el usuario espera que el analista adivine qué es lo que necesita, y este es un error que continúa presentándose, como se muestra en la fig. 6.8.4, en donde en el 54% de los proyectos se vio reflejado este indicador.

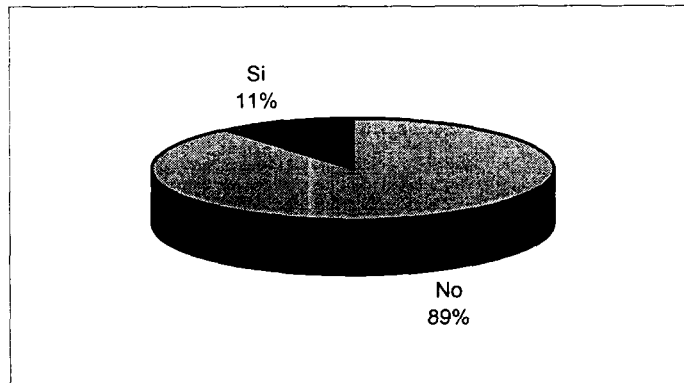


Fig. 6.8.5 No deseo del sistema

La fig. 6.8.5 muestra que algunos usuarios se ven forzados a utilizar alguna herramienta de trabajo que no desean, en el caso particular de los sistemas de información, este indicador se presenta en pocas ocasiones, ya que en la mayoría, los usuarios están abiertos a aceptar todo lo que pueda ayudarles a mejorar su rendimiento en el trabajo, y no se presenta una resistencia al cambio de manera significativa, aunque aún sigue existiendo este indicador.

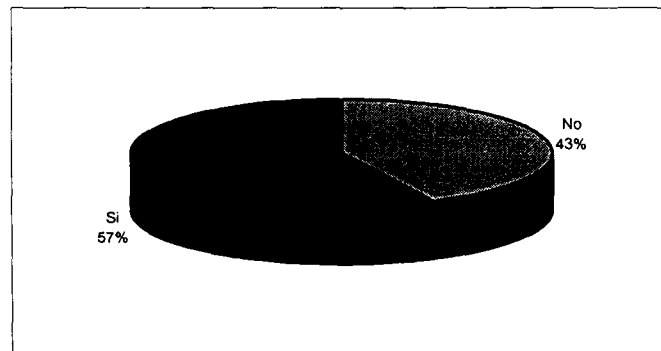


Fig. 6.8.6 Percepciones de las necesidades

La fig. 6.8.6 muestra que cuando se realiza un proyecto que será utilizado por más de un usuario, cada uno de ellos puede tener diferentes percepciones de los requerimientos de un sistema de información que les ayude a desarrollar sus funciones de una mejor forma. En el 57% de los casos, los usuarios diferencian en algún punto y esto ocasiona que los requerimientos lleguen a cambiarse cuando a alguno de estos usuarios no le satisface algún punto del sistema. En el 43% de los casos, a pesar de que el sistema se desarrolle para varios usuarios, se llega a tener un consenso general sobre las reales necesidades que se tienen.

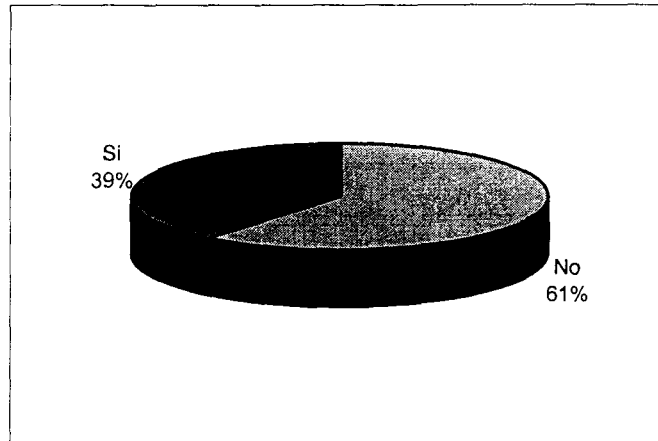


Fig. 6.8.7 Presiones políticas

Todo proyecto necesita tener un directivo que lo apoye y lo patrocine para que pueda ser llevado a cabo. En este sentido, cuando a un directivo no le parece algo que se está desarrollando con el proyecto, de alguna manera ejerce su posición dentro de la organización para pedirle los cambios necesarios al líder del proyecto. La fig. 6.8.7 muestra que los directivos ejercieron su autoridad en el 39% de los proyectos realizados, en tanto en el 61% restante, no se ejerce ninguna presión por parte de los mismos dejando al equipo de trabajo desarrollar sus actividades de manera independiente para lograr obtener el proyecto.

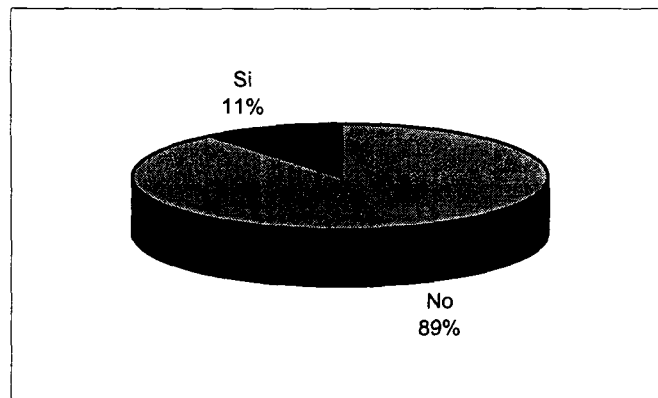


Fig. 6.8.8 Enfoque tecnológico débil

La fig. 6.8.8 muestra que el enfoque tecnológico que se proyecta para poder desarrollar un proyecto de desarrollo de sistemas de información, en su gran mayoría se realiza de manera correcta, y únicamente en el 11% de los casos, se presenta esta situación. Esto quiere decir que al momento de realizar el análisis de las necesidades del usuario, generalmente se proyecta adecuadamente los requerimientos tecnológicos que este requiere.

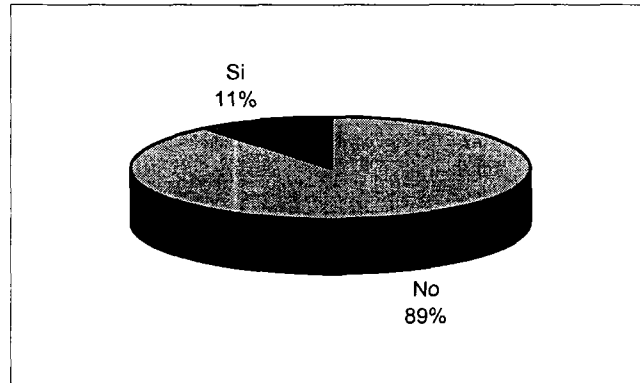


Fig. 6.8.9 Habilidades del equipo de trabajo

La fig. 6.8.9 muestra que en el 11% de los casos, las habilidades del equipo de trabajo no se desarrollan al máximo. Esto puede ser ocasionado por que el líder del proyecto no conoce a fondo las capacidades y alcances de sus elementos que conforman su equipo de trabajo, asignándoles tareas que tal vez no sean las adecuadas, o bien no se les capacita como debería de ser de acuerdo a sus funciones.

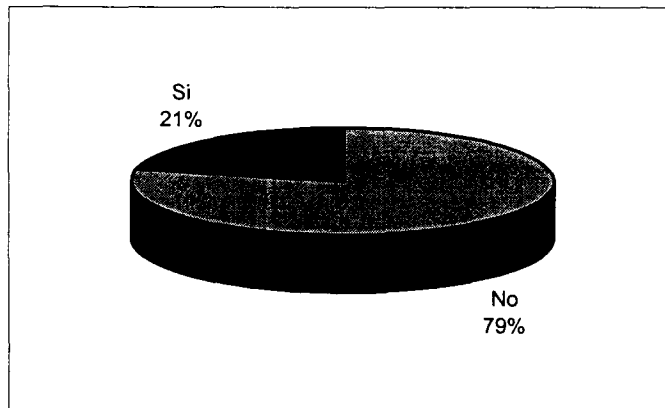


Fig. 6.8.10 Carencia de métricas de desempeño

La fig. 6.8.10 muestra que un factor importante en todo equipo de trabajo, es poder tener una forma de evaluar el desempeño de los elementos que conforman dicho equipo. En el 79% de los casos, los líderes de proyecto si tienen una relación de métricas que les ayudarán a evaluar el comportamiento de los integrantes de su equipo de trabajo. Esto le ayuda al líder de proyecto, a saber cómo reaccionan sus elementos en alguna condición determinada, a conocer un poco más sobre su efectividad y calidad en el desarrollo de sus actividades.

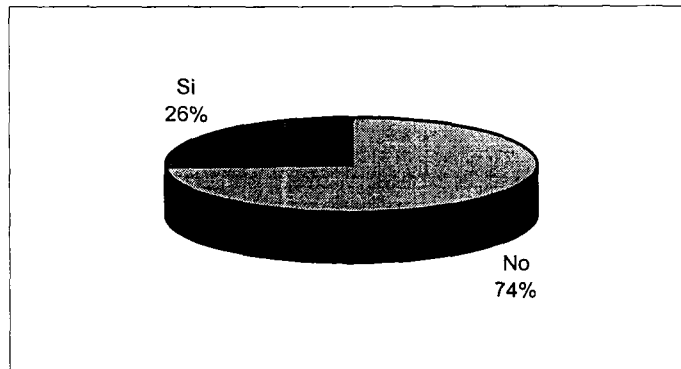


Fig. 6.8.11 Carencia de Apoyo directivo

Se mencionó anteriormente que el apoyo de un directivo en el proyecto es de gran importancia y ayuda para que éste pueda desarrollarse. Sin embargo, no en todos los casos se puede tener dicho apoyo. La fig. 6.8.11 muestra que en el 26% de los casos no se tiene apoyo de parte de los directivos, ocasionando que el proyecto esté a la deriva y únicamente con el sustento del líder del proyecto.

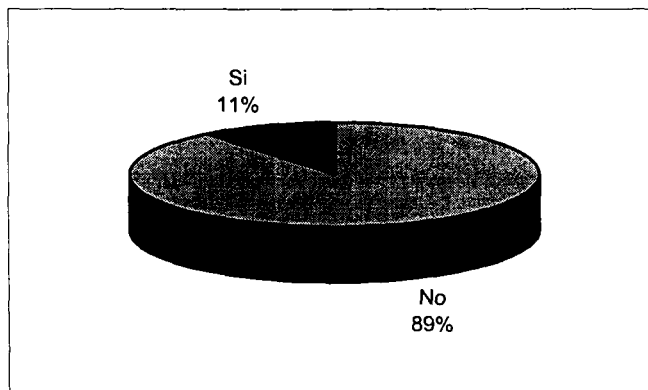


Fig. 6.8.12 Ruido en el área de desarrollo

La concentración y tranquilidad en un área de desarrollo de sistemas de información es de gran importancia, puesto que mantiene relajado y sin presiones externas al equipo de trabajo. En este sentido, la fig. 6.8.12 muestra que en la gran mayoría de los casos (89% del total) no se presentan ruidos provenientes del exterior al área de desarrollo, por lo tanto los elementos del equipo de trabajo no tienen distracciones y se concentran mejor para desempeñar sus labores.

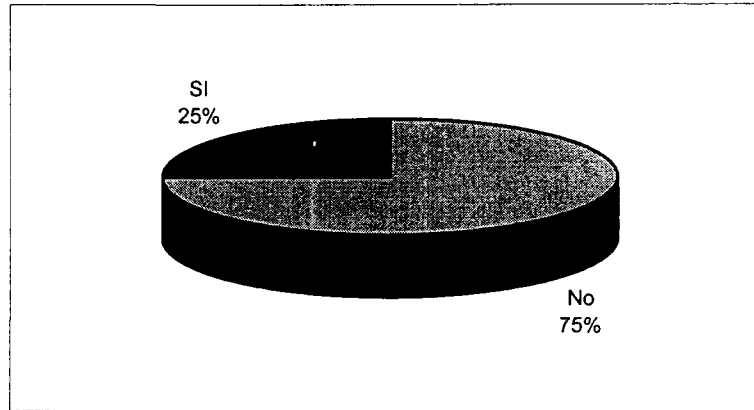


Fig. 6.8.13 Eliminación de pruebas de calidad

Las pruebas de calidad de los sistemas desarrollados, conforman uno de los últimos pasos antes de liberar dichos sistemas. Estas pruebas ayudarán a comprobar que el sistema funciona correctamente en todos y cada uno de los módulos que lo comprenden. La eliminación de estas pruebas, ocasiona que el usuario no trabaje con la confianza requerida, ya que no sabrá si obtendrá los resultados deseados. La fig. 6.8.13 muestra que se están eliminando únicamente en el 25% de los casos, y son producto de presiones del tiempo de entrega del proyecto.

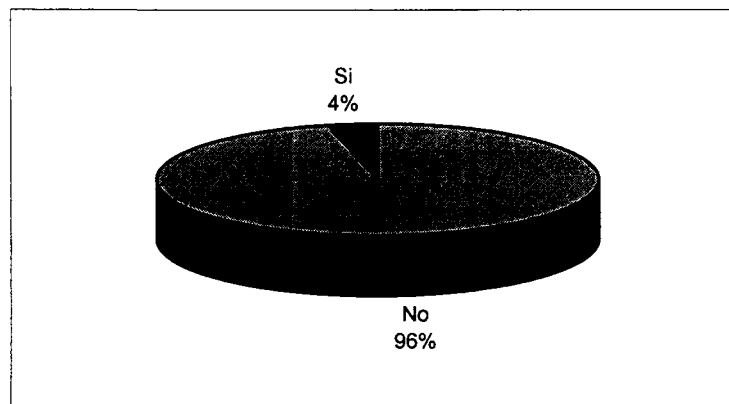


Fig. 6.8.14 No existe relación con la meta y visión de la empresa

Las funciones de los departamentos de sistemas, como lo muestra la fig. 6.8.14, tiene en su gran mayoría (96% del total) una relación con la meta y visión que persigue la compañía a nivel global. Esto indica que la empresa no considera al Depto. de Informática como un ente aislado a la empresa, sino como una parte integral del funcionamiento de la misma, ocasionando que los empleados se comprometan con su trabajo.

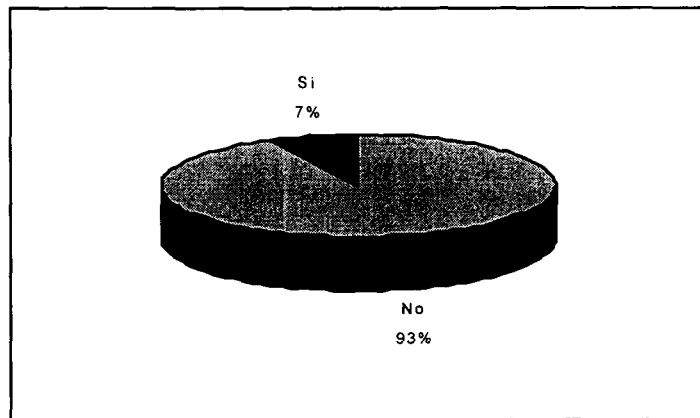


Fig. 6.8.15 Responsable directo del proyecto

La fig. 6.8.15 muestra que en el 93% de los casos, se tiene un responsable directo del proyecto, es decir, no es una responsabilidad única del equipo de desarrollo, sino que la responsabilidad del mismo cae sobre el líder del proyecto. Claro está que esto no indica que el equipo no trabaje en forma responsable, simplemente que la persona que está a cargo del proyecto es el líder del proyecto. En el 7% del total de los casos, la responsabilidad es compartida y no únicamente del líder del proyecto.

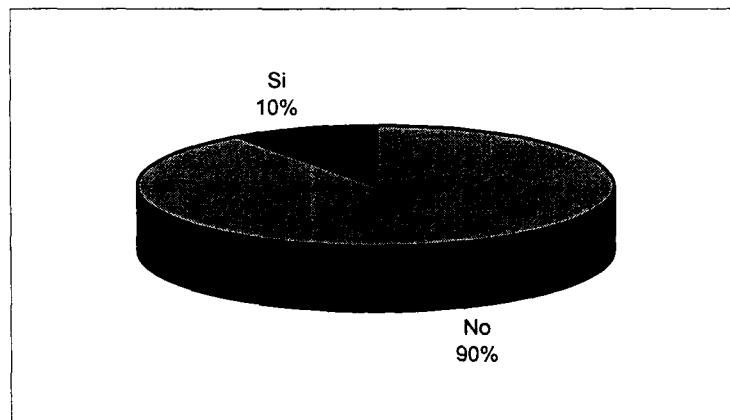


Fig. 6.8.16 Estructura de desarrollo débil

La fig. 6.8.16 muestra que el 90% de los proyectos contaron con todos los elementos necesarios para poder desarrollarlo. Únicamente el 10% de los proyectos no contaban con los suficientes recursos para poder realizar el proyecto, por lo que esto puede ocasionar desviaciones en el presupuesto, y sobre todo en el tiempo programado para su entrega.

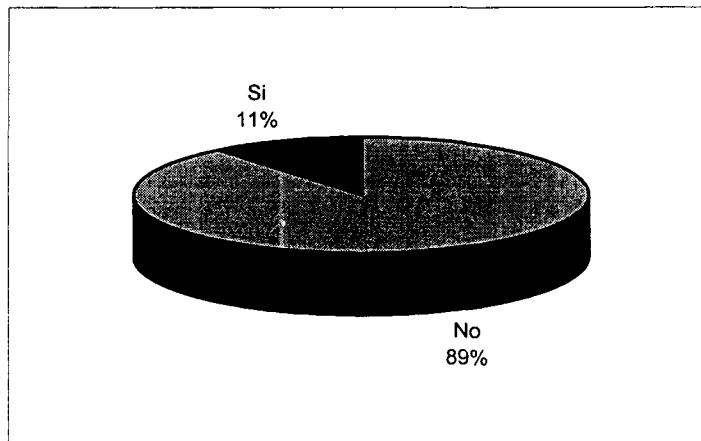


Fig. 6.8.17 Duplicidad de funciones

Una adecuada administración de las funciones de los elementos del equipo de trabajo es indispensable, para evitar que dos o más personas estén realizando las mismas actividades. La fig. 6.8.17 muestra que en el 11% del total de los casos, se encontró que se presenta esta situación de duplicidad de funciones, esto quiere decir que el líder del proyecto no está realizando sus funciones de manera correcta, ya que les está asignando las mismas funciones a más de una persona. Esto tiene como consecuencia el desaprovechamiento del recurso humano en primer lugar, además de otros factores que tengan relación con las actividades que realizan estas personas.

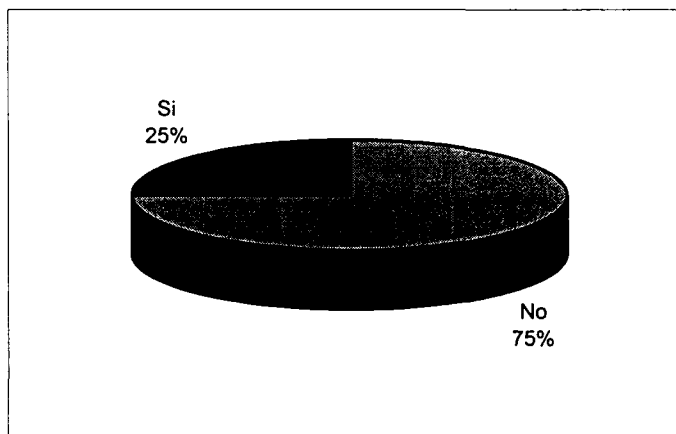


Fig. 6.8.18 Análisis de riesgo deficiente

La fig. 6.8.18 muestra, que únicamente el 25% de los casos tiene un análisis de riesgo mal desarrollado, y por lo tanto, no se tiene la confiabilidad de que al desarrollar el proyecto, este vaya a tener el resultado esperado. La mayor parte de los líderes de proyecto sí llevan a cabo su análisis de riesgo en forma adecuada, y esto les permite determinar antes de comenzar, cuál es el porcentaje de éxito del proyecto.

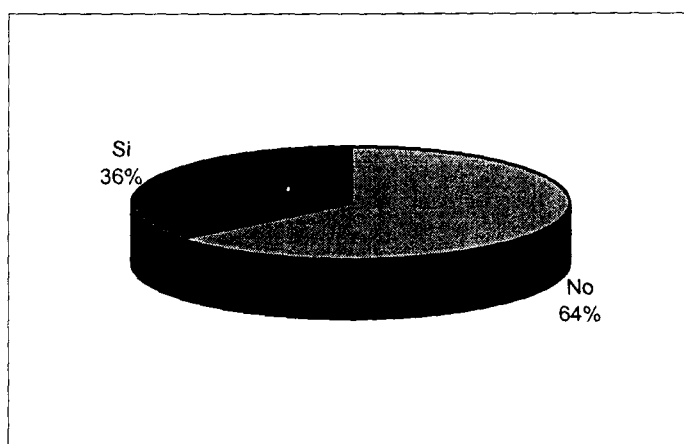


Fig. 6.8.19 Revisión de proyectos anteriores

La fig. 6.8.19 muestra que el 36% de los líderes de proyectos no realizan una revisión de los proyectos que llevaron a cabo anteriormente, con el fin de ubicar posibles indicadores que se presentaron para de esta manera asegurarse de que no se vayan a presentar en el proyecto que desarrollan actualmente. Por otro lado el 64% de los líderes de proyecto, sí realiza una revisión de proyectos pasados. Esta práctica puede darle una mayor tranquilidad al líder del proyecto al saber qué cosas fallaron, ya que pondrá atención especial para que no vuelvan a presentarse.

6.9 Origen de las fallas

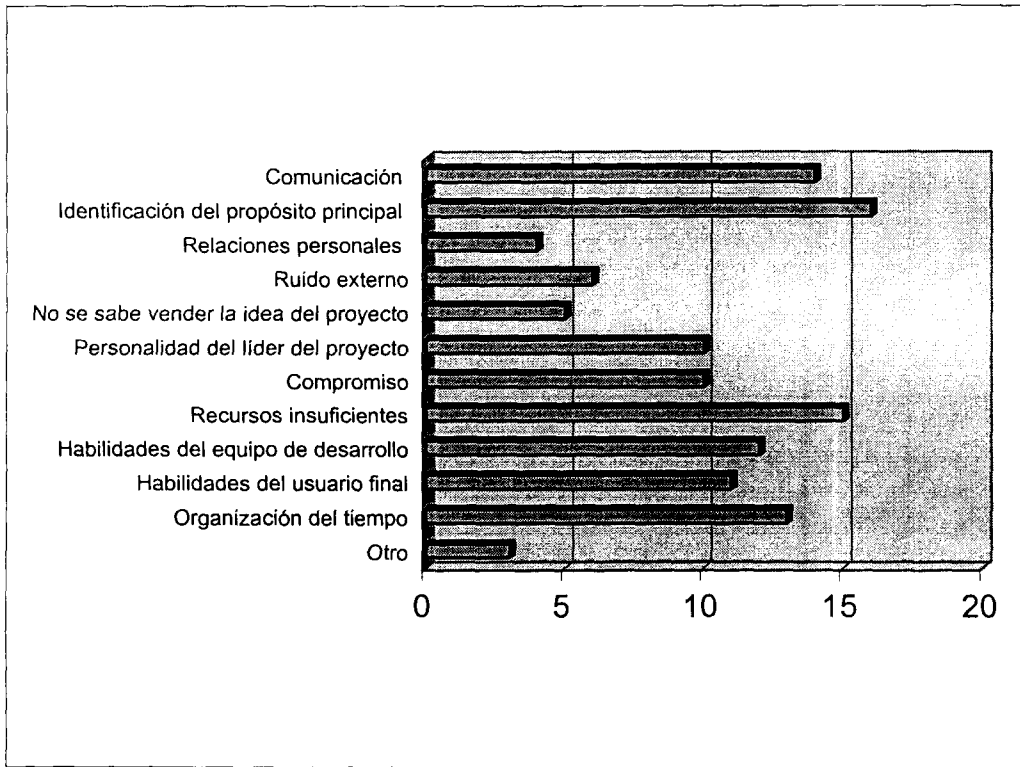


Fig. 6.9 Origen de las fallas

La fig. 6.9 muestra lo que a consideración de los diferentes líderes de proyectos son las causas de que existan errores dentro del ciclo de vida de desarrollo de un sistema de información. Las cinco causas mencionadas con mayor frecuencia son:

- Identificación del propósito principal.
- Recursos Insuficientes.
- Comunicación.
- Organización del tiempo.
- Habilidades del equipo de desarrollo.

Esto nos hace ver que no únicamente la capacidad de trabajo de los elementos del equipo de trabajo es lo que puede asegurar un éxito en un proyecto. De hecho no es la causa más citada por los líderes de proyecto. Las principales causas son enfocadas a factores ajenos a los elementos del equipo de trabajo, y tienen que ver más que nada con las actividades y responsabilidades del líder del proyecto, ya que es el encargado de proporcionar todas las herramientas de trabajo, de determinar qué es lo que desea el usuario

y de establecer un plan de trabajo adecuado. Un aspecto importante que sí involucra a todos los elementos del equipo de trabajo y no solamente al líder del proyecto es la comunicación entre ellos mismos y hacia el usuario.

6.10 Areas de mejora

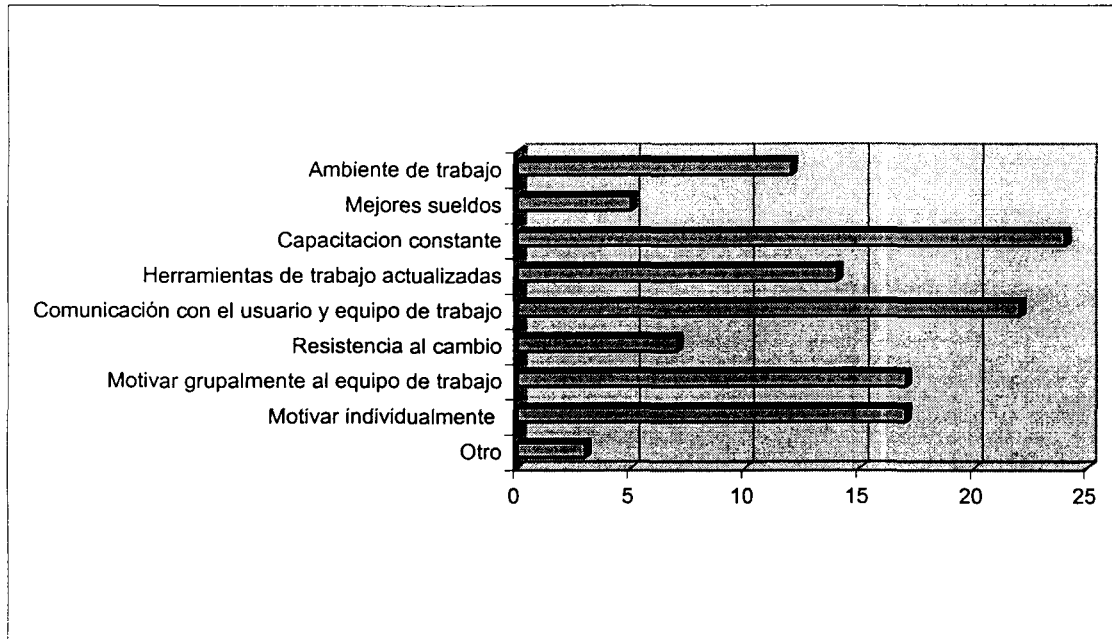


Fig. 6.10 Areas de mejora

La fig. 6.10 muestra las áreas de mejora que a criterio de los líderes de proyecto son las más importantes a evaluar y mejorar dentro del desarrollo de un proyecto, y no únicamente de uno relacionado con los sistemas de información. Las cinco áreas con más preocupación de los líderes de proyecto son:

- Capacitación constante.
- Comunicación con el usuario y equipo de trabajo.
- Motivación a nivel grupal.
- Motivación a nivel individual.
- Herramientas de trabajo actualizadas.

El mejoramiento de todas éstas áreas, dará a los líderes de proyecto, un mejor ambiente de trabajo, contara con personal más comprometido con los proyectos y tendrán una disponibilidad de trabajar siempre, además de tener personal con conocimientos en las últimas tecnologías que le brindarán un mejor servicio a los usuarios que soliciten el desarrollo de un nuevo sistema de información.

6.11 Conclusiones

En base a los objetivos establecidos en el capítulo 5, se formularon las preguntas de la encuesta buscando encontrar precisamente los puntos deseados para comprobar la teoría establecida en el marco teórico en comparación con la realidad en las empresas seleccionadas. Las preguntas cubrieron los puntos de comunicación en el trabajo de equipo, la disponibilidad de las personas, metodologías de trabajo, liderazgo, etc.

La mayor parte de los temas tratados en el marco teórico, se comprobaron en la investigación de campo. El punto más rescatable, y que no se trato a fondo en el marco teórico, fue el de las áreas de mejora que propusieron los diferentes líderes de proyecto que contestaron la encuesta. Estas áreas son de gran importancia no solamente para el desarrollo de un sistema de información, si no para todo trabajo en equipo, por lo tanto es adecuado que se realicen actividades encaminadas a cubrir dichas áreas para beneficio de los grupos de trabajo y por ende en las organizaciones.

Capítulo 7

Conclusiones y trabajos futuros

La estructura de un proyecto de desarrollo de sistemas de información está conformado por varias etapas. Esta tesis se desarrolló enfocada hacia el ciclo de vida de desarrollo, el cual es una de las más importantes dentro del entorno completo de desarrollo de un proyecto de estas características.

La encuesta utilizada como herramienta para la recolección de datos se diseñó con la idea de buscar en los líderes de proyectos, aquellos aspectos que consideran importantes en el trabajo en equipo y la comunicación que debe de existir en un entorno de trabajo conjunto, así como buscar qué es lo que le está originando que se tengan problemas con los equipos de trabajo o con el proyecto, y algo importante, el determinar cuáles son las áreas de mejora más importantes.

En lo referente a las metodologías de desarrollo que se utilizan actualmente en las empresas seleccionadas, encontramos que entre las que más prefieren los líderes de proyecto son la Entrega por Etapas y el Prototipo Evolutivo. Esto quiere decir, que los equipos de desarrollo, encabezados por su líder de proyecto, van creando poco a poco los diferentes módulos de los cuáles se integrara el futuro sistema de información. Estas prácticas, tienen la ventaja, de que pueden hacerse las modificaciones al sistema, al momento de detectar alguna nueva necesidad, o bien modificar algún modulo que ya esté en construcción, de esta manera se busca que los cambios que puedan surgir por parte del usuario en cuanto a las funciones que deba realizar el sistema de información no afecte en gran medida al diseño que ya se tenga de acuerdo a los requerimientos iniciales del usuario.

Dentro del proceso de desarrollo del proyecto, el usuario tiene una participación activa durante varias etapas del ciclo de vida, ya que es incluido en prácticamente todas las actividades, claro que no en todos los casos encontramos los mismos comportamientos, ya que en algunas empresas no los consideran para las pruebas y mantenimiento por mencionar algunas etapas, en otras empresas los consideran incluso en la codificación, por mas extraño que parezca.

La participación directa del usuario, puede verse como una ventaja en el sentido de que el sistema hará exactamente lo que el usuario desea, ya que al estar involucrado en el proyecto, puede darse cuenta cuando algo que desea no se está diseñando de forma correcta de tal forma que satisfaga sus necesidades, o bien puede darse el lujo de incrementar sus requerimientos, al final de cuentas el sistema será para él y el equipo de desarrollo tiene que hacer los cambios que el solicite.

Pero la participación directa del usuario no es del todo bien vista por algunos líderes de proyecto, tan es así que algunos los limitan exclusivamente a la etapa de análisis de requerimientos. La constante participación del usuario, y los cambios constantes de requerimientos afectarán el tiempo de desarrollo del proyecto, provocando que los tiempos de desarrollo y de entrega se vean afectados y se incrementen de acuerdo a la magnitud de los cambios requeridos por el usuario.

La comunicación es uno de los factores más importantes en los trabajos colaborativos en grupo. En este sentido, los líderes de proyecto no tienen inconveniente en la forma en cómo sus usuarios le realizan una petición para desarrollar un nuevo sistema de información. Igual atienden a un usuario que hace uso del software de mensajes instantáneos, que a un usuario que programa una cita especial para platicar sobre sus necesidades. Pero es importante notar que la gran parte de estos líderes de proyecto, una vez que se contactan por primera vez con el usuario, esta claro que se deberá de realizar una entrevista formal, en donde pueda o no pueda estar presente el jefe inmediato del usuario.

Esto se realiza para poder establecer una forma de trabajo más formal, y conocer al usuario cara a cara, y conocer sus inquietudes, sus reacciones, para que de alguna manera, el líder del proyecto, se vaya formando una visión de comportamiento del usuario, y saber de qué forma, es más fácil poder trabajar con el y obtener la mayor cantidad de información para realizar el análisis de sus requerimientos.

El líder del proyecto en su mayoría tiene programadas juntas con su personal del equipo de trabajo, con el fin de realizar una retroalimentación efectiva sobre sus actividades y realizar valoraciones sobre sus actividades correspondientes a determinado proyecto en el cuál se está trabajando.

El fin principal de la tesis era buscar aquellas causas, que originaban problemas en el tiempo del desarrollo de los proyectos. En este sentido, se identificaron varias causas, entre ellas la comunicación, la falta de herramientas necesarias para poder trabajar, las habilidades de los elementos que conforman el equipo de trabajo, ruidos en el área de trabajo, relaciones personales entre otras. La que más frecuentemente se presenta en estos equipos de trabajo es la identificación del propósito principal del proyecto, y esto tiene que ver mucho con la forma de comunicarnos con el usuario.

Al parecer, aún no encontramos la forma adecuada de tratar al usuario, y de obtener de este último, toda la información necesaria sobre sus necesidades. Tal vez un punto clave en esta situación, es que el usuario no sabe externar qué es lo que necesita, y no precisamente que no sepa que es lo que desea. Hay que hacer notar que no todos los usuarios piensan igual, y que todos tienen su

personalidad y su forma de expresarse de manera muy diferente a los demás. Es por eso que este es uno de los puntos más importantes antes de comenzar un proyecto, y si no se realiza de manera efectiva, el proyecto seguramente tendrá cambios en el futuro, o lo que es peor, podría cancelarse.

El anexo 7 muestra un mapa conceptual de los posibles errores y aciertos en los cuáles se puede caer al trabajar en un nuevo proyecto de desarrollo de sistemas de información. Todo comienza con el Cliente (usuario), si no tenemos un cliente, no tenemos un sistema a desarrollar, puesto que no tendremos quien esté interesado en el mismo, quien lo utilice, y si esto ocurre, no tiene ningún sentido desarrollar un nuevo proyecto. El mapa es simple de seguir, e ilustra en cada camino, las tareas o procesos que se realizan al tomar un camino, y los problemas o repercusiones que se presentan si se toma otro camino.

Trabajos futuros.

En el desarrollo de esta tesis, se pudo observar que existen varias alternativas a desarrollar como parte de trabajos subsecuentes a la misma. Para realizar trabajos referentes al área de desarrollo de sistemas de información, recomiendo que dichos trabajos, se enfoquen en descubrir los siguientes puntos:

- Impacto de la cancelación de un proyecto en el equipo de trabajo
- Motivación y compromiso del equipo de trabajo hacia el proyecto
- Diseñar indicadores de desempeño
- Establecer metodologías para buscar mejorar alguna de las áreas propuestas por los líderes de proyecto actuales.

Las dos primeras recomendaciones están orientadas a analizar, qué sentimientos son causados en el equipo de trabajo cuando un proyecto, le es cancelado. Básicamente es descubrir que tan importante es para el equipo de trabajo el cumplir o no con un proyecto, ¿sienten una frustración por no completar el proyecto?, ¿se presenta una condición de bajo rendimiento en el trabajo de proyectos futuros?. Preguntas como esta son las que habría que responder, esto ayudará a saber en que magnitud, el equipo de desarrollo siente como suyos los proyectos en los cuales participa, si en verdad les importa o no el que un proyecto llegue a su fin como debe de ser.

La tercera recomendación, está orientada a establecer algún modelo de rendimiento de los elementos del equipo de trabajo, buscando medir de alguna manera las capacidades individuales y grupales tanto individuales como para trabajar en equipo. Algunos líderes de proyecto carecen de este tipo de indicadores, por lo tanto no saben cómo evaluar a los elementos de sus equipos de trabajo. Esto ayudará a determinar y saber bajo qué condiciones son más

productivos y eficientes, saber bajo qué condiciones la calidad en el trabajo disminuye, saber cuál es el número más adecuado para el tamaño del equipo, etc.

El último apartado de recomendaciones, se enfoca en buscar alguna metodología que permita a los líderes de proyecto, o en su caso a los Directores de Informática, establecer programas de mejora para las áreas como sueldos, capacitación, crecimiento de los empleados dentro de la empresa, etc., y en general aquellos puntos indicados en la fig.6.9 que son considerados como importantes por los actuales líderes de proyecto. Esto tiene como finalidad, que se conozca más a fondo a los elementos de los equipos de trabajo, conocerlos en sus intereses tanto personales, como profesionales, de esta forma, puede formarse un perfil de los miembros del equipo de trabajo, y se pueden tomar acciones encaminadas a incrementar su motivación tanto personal como profesional, y de esta forma, evitar que se distraiga en asuntos ajenos a su proyecto, evitar que pueda interesarse en irse a otra compañía, y a la vez ir formando algún nivel de lealtad y compromiso hacia su equipo de trabajo y por ende a la compañía. Este puede ser una opción para evitar por ejemplo, las inconformidades de los empleados, o simplemente, asegurarse de que se evitará en alguna medida la rotación de personal, que finalmente se traduce en problemas para los líderes de proyecto, ya que tienen que reemplazar a algún elemento involucrando tiempo para seleccionarlo, capacitación, etc..

Anexo 1 Instrumento de la investigación

Instituto Tecnológico y de Estudios Superiores de Monterrey
Encuesta para la Investigación de Tesis en la
Maestría de Administración de Tecnologías de Información

"Indicadores de falla presentados en el ciclo de vida
de un sistema de información"

Propuesta por Eduardo Samaniego Hernández, LI

Nota: La información obtenida en la presente investigación será usada sólo con fines académicos para el desarrollo de la tesis en cuestión, guardando íntegramente confidencialidad en los datos que sean aportados. Su correo electrónico y/o número telefónico se solicitan únicamente para enviarle una copia de los resultados de la investigación y aclarar las dudas que puedan surgir; no se divulgarán ni se liberarán a terceros. Agradezco de antemano su colaboración en este proyecto. Mil gracias.

1. Cuando inicia el desarrollo de un nuevo sistema de información, ¿cuál es el criterio que más se acerca para seleccionar al personal y asignarle el trabajo?

- El que este libre. (6%)
- Por experiencia y dominio de un tema. (65%)
- Por equipo de trabajo. (16%)
- Por turno (lleva un rol de quien es el siguiente para aceptar un proyecto).(0%)
- Otro (Especifique): (13%) _____

2. De acuerdo al servicio que se presta a todas las áreas de la empresa, indique el porcentaje que más se acerque a los siguientes tópicos, tomando como base la cantidad total de sistemas de información que se desarrollaron en el último año:

Los sistemas se entregaron en la fecha establecida y sin demora alguna:

- 0 – 25 % (20%)
- 26 – 50 % (20%)
- 51 _ 75 % (33%)
- 76 _ 100 % (27%)

Se hacen modificaciones a los requerimientos iniciales una vez que se a comenzado el proyecto:

- 0 – 25 % (10%)
- 26 – 50 % (37%)
- 51 _ 75 % (23%)
- 76 _ 100 % (30%)

Los sistemas cumplen con las expectativas de los usuarios:

- 0 – 25 % (0%)
- 26 – 50 % (3%)
- 51 _ 75 % (27%)
- 76 _ 100 % (70%)

Los sistemas se usan adecuadamente gracias a la capacitación sobre su uso:

0 – 25 % (3%) 26 – 50 % (7%) 51 _ 75 % (30%) 76 _ 100 % (60%)

Se presentó al menos una situación adversa que originó elevar el presupuesto del proyecto, modificar el plan de trabajo o cancelarlo:

0 – 25 % (36%) 26 – 50 % (37%) 51 _ 75 % (20%) 76 _ 100 % (20%)

Se presentaron de 2 a 5 situaciones adversas que originaron elevar el presupuesto del proyecto, modificar el plan de trabajo o cancelarlo:

0 – 25 % (64%) 26 – 50 % (25%) 51 _ 75 % (11%) 76 _ 100 % (0%)

Se presentaron de 6 a 10 situaciones adversas que originaron elevar el presupuesto del proyecto, modificar el plan de trabajo o cancelarlo:

0 – 25 % (93%) 26 – 50 % (7%) 51 _ 75 % (0%) 76 _ 100 % (0%)

Los problemas presentados en el desarrollo fueron originados por la tecnología de informática (Hardware y/o Software) que se utilizó:

0 – 25 % (80%) 26 – 50 % (17%) 51 _ 75 % (3%) 76 _ 100 % (0%)

Los problemas presentados en el desarrollo fueron originados por la falta de habilidades del equipo:

0 – 25 % (77%) 26 – 50 % (20%) 51 _ 75 % (0%) 76 _ 100 % (3%)

Los problemas presentados en el desarrollo fueron de carácter meramente económicos:

0 – 25 % (77%) 26 – 50 % (7%) 51 _ 75 % (13%) 76 _ 100 % (3%)

Los problemas presentados en el desarrollo fueron por falta de personal:

0 – 25 % (73%) 26 – 50 % (17%) 51 _ 75 % (7%) 76 _ 100 % (3%)

3. En los siguientes reactivos, indique el rango en que está más acorde a lo definido, en donde "1" significa que está en total desacuerdo, y "5" indica que está totalmente de acuerdo, NA indica que no aplica a ese criterio. En cada una de las siguientes etapas de desarrollo de sistemas de información, ¿se cuenta con la participación directa del usuario?

			NA	1	2	3	4	5
3.1 Identificación de necesidades (determinación de requerimientos).	No 14%	Si 86%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3.2 Estudio de factibilidad (económica, humana, técnica, operativa, etc).	No 3%	Si 57%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3.3 Análisis.	No 33%	Si 67%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3.4 Diseño Lógico.	No 75%	Si 25%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3.5 Diseño Físico.	No 79%	Si 21%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3.6 Codificación.	No 89%	Si 11%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3.7 Pruebas.	No 36%	Si 64%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3.8 Instalación.	No 57%	Si 43%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
3.9 Mantenimiento.	No 50%	Si 50%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

4. En los siguientes reactivos, indique el rango en que está más acorde a lo definido, en donde "1" significa que está en total desacuerdo, y "5" indica que está totalmente de acuerdo. NA indica que no aplica a ese criterio. En el desarrollo del nuevo sistema de información:

			NA	1	2	3	4	5
4.1 Se utilizan metodologías de desarrollo previamente establecidas (clásico, en cascada, prototipo, espiral, etc.).	No 22%	Si 78%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4.2 Las actividades a desarrollar se definen una vez que se tiene seleccionada la metodología de desarrollo más conveniente de acuerdo a las características del proyecto.	No 36 %	Si 64%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4.3 La metodología de desarrollo es evaluada y seleccionada por el líder de proyecto.	No 48%	Si 52%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4.4 Los miembros del equipo de desarrollo conocen las fases en que se compone la metodología de desarrollo.	No 32%	Si 68%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
4.5 Se tienen las herramientas necesarias para desarrollar el sistema de información (software de base de datos, comunicaciones, hardware, personal, etc).	No 18%	Si 82%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

5. Cuando el usuario desea el desarrollo de un nuevo sistema de información ¿cómo le hace saber sus necesidades iniciales para dicho sistema?

- Por escrito (carta, memorándum, aviso, etc.). (18%)
- En una entrevista en su oficina. (32%)
- Por correo electrónico, ICQ, Yahoo Messenger, etc. (13%)
- Mediante una reunión con el y su jefe inmediato. (24%)
- Otro (Especifique): (13%) _____

6. ¿Existe algún reglamento interno en el Depto. de Informática en donde se especifica la Misión de dicho departamento?

- Si, ¿Tiene relación directa con las metas de la organización? (72%)
 Si (100%) No (0%)
 No (28%)

7. En los siguientes reactivos, indique el rango en que está más acorde a lo definido, en donde "1" significa que está en total desacuerdo, y "5" indica que está totalmente de acuerdo. NA indica que no aplica a ese criterio.

		NA	1	2	3	4	5
7.1 Las decisiones de la Alta Gerencia que involucran al Depto. de Informática son transmitidas a usted en forma oportuna. <i>No 39% Si 61%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7.2 Existen canales bien definidos de comunicación dentro del Depto de Informática, y con los demás Deptos. de la empresa. <i>No 43% Si 57%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7.3 El equipo de trabajo muestra una amplia disponibilidad para realizar sus labores. <i>No 14% Si 86%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7.4 El equipo de trabajo necesita que sea revisado su trabajo por el Líder del Proyecto antes de liberarlo. <i>No 29% Si 71%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7.5 El equipo de trabajo comprueba la compatibilidad de su trabajo con las metas establecidas para el proyecto. <i>No 18% Si 82%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7.6 Al terminar sus actividades, algún miembro del equipo participa en alguna otra actividad perteneciente a algún otro miembro del equipo de trabajo que aún no termina lo que le corresponde. <i>No 25% Si 75%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7.7 Cada miembro del equipo de trabajo resuelve por sí mismo los problemas de trabajo que le presenta el proyecto. <i>No 71% Si 29%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7.8 El Líder del Proyecto se encarga de administrar únicamente al equipo de trabajo y sus actividades correspondientes, sin participar en el análisis, diseño o codificación del proyecto que se desarrolla. <i>No 93% Si 7%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7.9 El Líder del Proyecto es quien decide la metodología de desarrollo a seguir. <i>No 50% Si 50%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7.10 El Líder del Proyecto es quien define las actividades en cada una de las etapas de la metodología seleccionada. <i>No 30% Si 70%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7.11 Los miembros del equipo de trabajo son quienes deciden qué plataforma de desarrollo utilizar, qué software y equipo requiere el sistema en desarrollo, y en general de tomar las decisiones en cuanto a tecnología que se requiera para el proyecto. <i>No 89% Si 11%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

	NA	1	2	3	4	5
7.12 El Líder del Proyecto y/o Gerente de Sistemas autorizan los presupuestos del proyecto. <i>No 50% Si 50%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
7.13 Al término del proyecto, el usuario emplea parte de su tiempo de trabajo en conocer el sistema de información que acaban de entregarle. <i>No 21% Si 79%</i>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

8. La retroalimentación hacia el equipo de trabajo, se presenta:

- Cuando existen dudas en la mayoría del personal. (11%)
- Periódicamente (semanal, mensual, etc). (63%)
- Cuando un miembro del equipo no sabe exactamente lo que debe de hacer. (20%)
- Otro (Especifique) (6%) _____

9. ¿Tiene alguna metodología de desarrollo favorita y que sea la que prevalece en la mayoría de los proyectos?

- Si (70%)
- No (30%)

10. Si la respuesta anterior es "SI", indique cuál de las siguientes metodologías de desarrollo es la que concuerda con su respuesta. Si la respuesta es "NO", pase a la pregunta 11.

- Clásico. (24%)
- Cascada Pura. (0%)
- Modificar y Corregir. (6%)
- Espiral. (4%)
- Sashimi (Cascada con Fases Solapadas). (0%)
- Subproyectos. (4%)
- Prototipo Evolutivo. (20%)
- Entrega por etapas. (36%)
- Otra (Especifique): (12%) _____

11. En los siguientes reactivos, indique el rango en que está más acorde a lo definido, en donde "1" significa que está en total desacuerdo, y "5" indica que está totalmente de acuerdo. NA indica que no aplica a ese criterio. De las siguientes situaciones que pueden presentarse en el desarrollo de un sistema de información y de acuerdo a su experiencia, seleccione el indicador o los indicadores que considere se presentan con mayor frecuencia en el desarrollo del sistema de información.

		NA	1	2	3	4	5
11.1 No se utiliza una metodología de desarrollo.	No 86% Si 14%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.2 Cambios en los requerimientos por parte del usuario.	No 42% Si 58%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.3 Las fechas de inicio y/o entrega de las etapas del ciclo de vida se traslapan o no se cumplen.	No 57% Si 43%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.4 El usuario no sabe externar sus necesidades.	No 54% Si 46%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.5 El usuario no desea al 100% el desarrollo de un nuevo sistema de información, es por imposición de su jefe.	No 89% Si 11%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.6 Diferentes percepciones de necesidades por parte de más de un usuario.	No 43% Si 57%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.7 Presiones políticas por parte de algún directivo.	No 61% Si 39%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.8 Enfoque tecnológico débil.	No 89% Si 11%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.9 Habilidades del equipo de trabajo no desarrolladas a su máximo potencial.	No 89% Si 11%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.10 Carencia de métricas de desempeño.	No 79% Si 21%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.11 No se tiene un directivo que apoye al proyecto.	No 74% Si 26%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.12 Existe ruido en el área de desarrollo del sistema de información que provoca malestar al equipo de desarrollo.	No 89% Si 11%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.13 Eliminación de pruebas de calidad por presión del tiempo.	No 75% Si 25%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.14 No existe relación con la meta y visión de la empresa.	No 96% Si 4%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.15 No existe un responsable directo del proyecto (no Directivo).	No 93% Si 7%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.16 Tiene una estructura de desarrollo débil (poco equipo físico, no se tiene el software de desarrollo adecuado, falla en las comunicaciones, falta de personal, etc).	No 90% Si 10%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.17 Duplicidad de funciones.	No 89% Si 11%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.18 Análisis de riesgo deficiente.	No 75% Si 25%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
11.19 No se revisa el desarrollo de proyectos anteriores para analizar las situaciones que hayan ocasionado su desvío y/o cancelación.	No 64% Si 36%	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

12. Conforme a su experiencia, ¿qué es lo que causa que un proyecto tenga problemas en su desarrollo? (Puede seleccionar varios)

- Comunicación entre los miembros del equipo de trabajo. (11.76%)
- Identificación del propósito principal del proyecto. (13.45%)
- Relaciones personales de los miembros del equipo de trabajo. (3.36%)
- Ruido externo en la empresa (presiones políticas, empresariales, gente externa a la organización, etc.). (5.04%)
- No se sabe vender la idea del proyecto. (4.20%)
- Personalidad del líder del proyecto. (8.40%)
- Compromiso con el proyecto y/o la organización. (8.40%)
- Recursos insuficientes (humanos, económicos, tecnológicos, etc.). (12.61%)
- Habilidades del equipo de desarrollo. (10.08%)
- Habilidades del usuario final (eficiencia en el uso del sistema, interés en el proyecto, etc.). (9.24%)
- Organización del tiempo para efectuar todas las tareas por parte del equipo de desarrollo. (10.92%)
- Otro (Especifique): (2.52%) _____

13. En el mismo sentido, ¿qué se debe de hacer para prevenir que alguna situación afecte al proyecto de manera tal que pueda provocar un retraso, carezca de recursos o sea cancelado? (Puede seleccionar varios)

- Mejorar el ambiente de trabajo. (9.92%)
- Mejores sueldos. (4.13%)
- Capacitación constante (usuarios y equipo de trabajo). (19.83%)
- Herramientas de trabajo actualizadas (software, computadoras, etc). (11.57%)
- Comunicación con el usuario y con el equipo de trabajo. (18.18%)
- Resistencia al cambio por parte del usuario. (5.79%)
- Motivar grupalmente al equipo de trabajo. (14.05%)
- Motivación individual de cada miembro del equipo de trabajo. (14.05%)
- Otro (Especifique): (2.48%) _____

Si desea recibir información sobre los resultados de esta investigación, favor de proporcionar los siguientes datos para su localización:

E-mail : _____

Teléfono : _____

Anexo 2

Variables de la Investigación

Apartado de Investigación	Variables contenidas
1. Criterio de asignación de nuevos proyectos	1.1 Libre 1.2 Experiencia 1.3 Por equipo 1.4 Por turno 1.5 Otro_criterio
2. Desarrollo de sistemas en el último año	2.1 Fecha_entrega 2.2 Modificaciones_iniciado 2.3 Cumple_expectativas 2.4 Uso_adequado 2.5 Una_situación 2.6 Dos_a_cinco 2.7 Seis_a_diez 2.8 Problemas_tecnologia 2.9 Problemas_habilidades 2.10 Problemas_económicos 2.11 Problemas_personal
3. Participación directa del usuario	3.1 Usuario_factibilidad 3.2 Usuario_identificación 3.3 Usuario_análisis 3.4 Usuario_lógico 3.5 Usuario_físico 3.6 Usuario_codificación 3.7 Usuario_pruebas 3.8 Usuario_instalación 3.9 Usuario_mantenimiento
4. Inicio del proyecto	4.1 Usa_metodología 4.2 Define_actividades 4.3 Evalúa_metodología 4.4 Conocen_fases 4.5 Herramientas_necesarias
5. Comunicación y retroalimentación	5.1 Escrito 5.2 Entrevista 5.3 Mensaje_electrónico 5.4 Reunión 5.5 Otro_requerimientos_usuario 6.1 Si_reglamento 6.2 No_reglamento

	<p>6.3 Si_metas 6.4 No_metas 7.1 Transmisión_alta_gerencia 7.2 Canales_otras 7.3 Disponibilidad_equipo 7.4 Revisar_equipo 7.5 Compatibilidad_metas 7.6 Miembro_participa 7.7 Miembro_resuelve 7.8 Líder_administra 7.9 Líder_decide 7.10 Líder_actividades 7.11 Equipo_decide 7.12 Líder_presupuesto 7.13 Usuario_capacita 8.1 Dudas_mayoría 8.2 Periódicamente 8.3 Desconoce_miembro 8.4 Otro_retroalimentación</p>
6. Metodologías de desarrollo	<p>9.1 Si_favorita 9.2 No_favorita 10.1 Clásico 10.2 Cascara_pura 10.3 Modificar_corregir 10.4 Espiral 10.5 Sashimi 10.6 Subproyectos 10.7 Prototipo 10.8 Etapas 10.9 Otra_metodología</p>
7. Indicadores de falla presentados en el último año en el desarrollo de sistemas de información	<p>11.1 No_metodología 11.2 Cambios_usuario 11.3 Fechas_incumplidas 11.4 Usuario_externar 11.5 Usuario_desea 11.6 Diferentes_percepciones 11.7 Presiones_políticas 11.8 Enfoque_débil 11.9 Habilidades_no_desarrolladas 11.10 Carencia_métricas 11.11 Apoyo_directivo 11.12 Area_ruido 11.13 Elimina_pruebas 11.14 Relación_misión_visión</p>

	<p>11.15 Responsable_proyecto 11.16 Estructura_débil 11.17 Duplicidad_funciones 11.18 Análisis_riesgo 11.19 Revisión_proyectos</p>
8. Origen de fallas	<p>12.1 Comunicación 12.2 Identificar_principal 12.3 Relaciones_personales 12.4 Ruido_externo 12.5 No_idea 12.6 Personalidad_líder 12.7 Compromiso 12.8 Recursos_insuficientes 12.8 Habilidades_equipo 12.9 Habilidades_usuario_final 12.10 Organizar_tiempo 12.11 Otro_causa</p>
9. Areas de mejora	<p>13.1 Mejor_ambiente 13.2 Mejores_sueldos 13.3 Capacitación_constante 13.4 Herramientas_actualizadas 13.5 Comunicación 13.6 Resistencia_cambio 13.7 Motivar_grupalmente 13.8 Motivar_individualmente 13.9 Otro_preveer</p>

Anexo 3

Test de supervivencia

El siguiente Test, ilustrará si el proyecto en el cuál se esta trabajando o evaluando, puede llegar a tener un fin deseado, o es necesario corregir algunas deficiencias. Asigna un 3 si la respuesta es SI, asigna un 2 si sientes que es probable que ocurra, y un 1 para aquel que no es posible.

Si el proyecto se encuentra en las etapas iniciales, contestar las preguntas de acuerdo a el plan del proyecto, si ya está asimilado, contesta las preguntas basandose en lo que está ocurriendo en el proyecto.

Requerimientos

El proyecto tiene una visión o misión clara y sin ambigüedades	___
¿Todos los miembros del equipo creen que la visión es realista?	___
¿ El proyecto es derivado de un caso especial de la empresa, que detalla los beneficios y cómo serán medidos esos beneficios?	___
¿El proyecto tiene un prototipo para demostrar las funciones que el sistema tendrá?	___
¿Tiene un especificación detallada y escrita de lo que se supone el sistema realizará?	___
¿El equipo entrevistó a la gente que trabaja actualmente con el sistema y continuan involucrados a través de todo el desarrollo del proyecto?	___
¿Tiene el proyecto un detallado y escrito plan de desarrollo de software?	___
¿La lista de tareas del proyecto incluye tales como instalación del programa, conversión de datos del sistema anterior, juntas con los clientes, etc?	___
¿La calendarización y presupuesto se actualizan al terminar cada fase del proyecto?	___
¿El proyecto tiene detallados y escritos los documentos de diseño y arquitectura?	___
¿Tiene detallado y escrito un plan de aseguramiento de calidad que requiere el diseño y revisión del código adicional a la prueba del sistema?	___
¿Se tiene detallado y escrito el plan de entrega del software indicando en donde sera entregado e instalado el sistema?	___
¿El plan del proyecto tiene contemplados días festivos, tiempo para días de enfermedad, entrenamieto, etc.?	___
¿El plan de trabajo fue aprobado por el equipo de desarrollo, el personal de aseguramiento de calidad y el personal técnico, en otras palabras, la gente acepta su responsabilidad en las cosas que hace?	___

Control del proyecto

¿Tiene a un ejecutivo clave de la empresa que tome las decisiones para hacerse el responsable del proyecto y da el soporte necesario al mismo?	___
¿Las funciones del administrador del proyecto le permiten dedicarle el tiempo adecuado al proyecto?	___
¿El proyecto tiene bien detallados los pequeños detalles que hacen que el proyecto este o no hecho al 100%?	___
¿Tiene el proyecto un canal de retroalimentación en donde los miembros del equipo de trabajo pueden anónimamente reportar problemas a sus administradores?	___
¿Tiene un plan escrito para control de cambios en las especificaciones del software?	___
¿Tiene el proyecto a alguien con la autoridad para aceptar o rechazar los cambios que se propongan?	___
¿El plan y estatus del proyecto están disponibles para todo el equipo de trabajo?	___

Administración de Riesgos

¿Tiene análisis de los riesgos actuales del proyecto? ¿se actualiza recientemente?	___
¿Tiene un responsable de identificar los riesgos emergentes del proyecto?	___
En caso de usar <i>outsourcing</i> , ¿tiene un plan para administrar a cada uno de ellos y una persona a cargo de cada uno? (3 si no usan)	___

Personal

¿Tiene el equipo la experiencia necesaria para completar el proyecto?	___
¿Tiene el equipo experiencia en el ambiente de trabajo donde será operado el sistema?	___
¿Tiene el proyecto un líder capaz de liderar el proyecto exitosamente?	___
¿Existe el personal necesario para hacer todo el trabajo requerido?	___
¿Trabajan bien en conjunto?	___
¿Está cada persona asignada al proyecto?	___

Total Preliminar (Suma de todos los puntos)

Multiplo

- 1.5 si tiene 3 o menos personal de tiempo completo
- 1.25 si tiene de 4 a 6
- 1 en otros casos

Total final. Multiplicar el total preliminar por el Multiplo

Resultado	Situación de tu Proyecto
90 o más	<p>Outstanding El proyecto está virtualmente garantizado al éxito en todos los aspectos, ya que se conocen bien la calendarización, el presupuesto, calidad, etc.</p>
80 – 89	<p>Excelente Un proyecto en este nivel esta desarrollandose mucho mejor que el promedio. el proyecto tiene una amplia posibilidad de entregar el proyecto en el tiempo, presupuesto y calidad esperados.</p>
60 – 79	<p>Bueno Representa mejor que el promedio de efectividad en el desarrollo de sistemas, Se conocen bien la calendarización o el presupuesto, pero e spoco probable que se conozcan ambos</p>
40 – 59	<p>Falla Es típico que se presente. Un proyecto con este resultado experimenta un alto stress en su equipo de trabajo y el software finalmente será entregado con menos funcionalidad que la deseada a un costo y en un tiempo mayor.</p>
< 40	<p>Riesgo Un proyecto con este resultado tiene debilidades significativas en la mayoría de sus áreas, ya sea en requerimientos, planeación, control, administración del riesgo o personal.</p>

Anexo 4

Tabla comparativa de Modelos de Desarrollo

Ventajas e inconvenientes de los modelos de ciclo de vida

Capacidades	1	2	3	4	5	6	7
A	**	*	*	****	** a ****	****	*
B	***	*	*	****	** a ****	* a **	*
C	**	****	*	****	****	**	****
D	*	****	* a **	****	****	****	****
E	**	*	*	****	**	**	**
F	**	**	*	**	**	*	**
G	*	*	****	**	****	**	**
H	**	*	* a ****	**	**	****	*
I	*	*	**	****	**	****	**
J	*	**	*	****	** a ****	**	****
K	**	**	****	*	* a **	*	**

1. Clásico

2. Cascada Pura

3. Codificar y Corregir

4. Espiral

5. Cascadas Modificadas

6. Prototipo Evolutivo

7. Entrega por Etapas

**** Excelente

*** Bueno

** Medio

* Malo

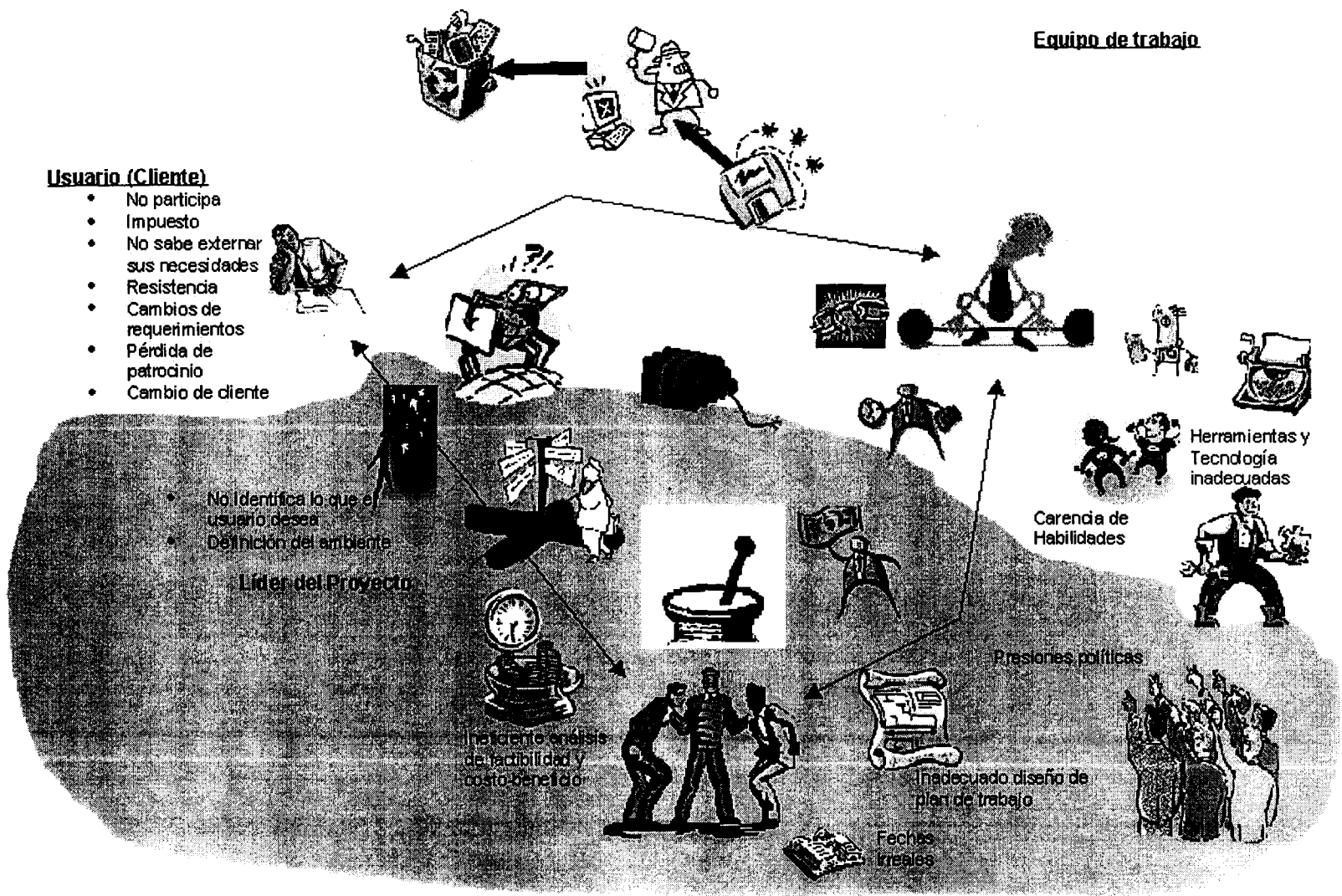
A. Trabaja con poca identificación de los requerimientos se refiere a lo bien que funciona el modelo de ciclo de vida cuando el cliente o usted identifican mal los requerimientos del sistema o cuando su cliente es propenso a modificar los requerimientos. Esto indica si el modelo es el adecuado para el desarrollo exploratorio del software.

B. Trabaja con poca comprensión sobre la arquitectura se refiere a lo bien que funciona el modelo de ciclo de vida cuando se está desarrollando en un área nueva de aplicación o en áreas de aplicación familiares pero con capacidades de desarrollo desconocidas.

- C. Genera un sistema altamente fiable** se refiere a los defectos que pueden aparecer en un sistema desarrollado con un modelo de ciclo de vida cuando comienza a funcionar
- D. Genera un sistema con amplio desarrollo** se refiere a la facilidad con la que se puede modificar el sistema en tamaño y diversidad durante su tiempo de vida. Esto incluye las modificaciones del sistema que no hayan sido incluidas por los diseñadores originales.
- E. Gestiona riesgos** se refiere a la capacidad del modelo para identificar y controlar los riesgos durante su planificación, riesgos para el producto, y otro tipo de riesgos.
- F. Estar sometido a una planificación predefinida** se refiere a si el modelo de ciclo de vida soporta una fecha de caída de rentabilidad inamovible para la entrega del software.
- G. Requiere poco tiempo de gestión** se refiere a la cantidad de tiempo de gestión y técnico que se requiere para utilizar el modelo de forma efectiva. Este tiempo incluye planificación, seguimiento del estado, producción de la documentación, adquisición de paquetes y otras actividades que no se incluyen directamente en la producción del software.
- H. Permite modificaciones a medio camino** se refiere a la capacidad para modificar aspectos significativos del producto a mitad del camino de la planificación del desarrollo. Esto no incluye la modificación de la función principal del producto, pero si incluye la extensión significativa de la misma.
- I. Ofrece a los clientes signos visibles de progreso** se refiere a cómo el modelo genera automáticamente signos de progreso que el cliente puede utilizar para realizar un seguimiento del estado del proyecto.
- J. Ofrece a la directiva signos visibles de progreso** se refiere a si el modelo genera automáticamente signos visibles que la directiva puede utilizar para seguir el estado del proyecto.
- K. Requiere poca sofisticación para la directiva y desarrolladores** se refiere a especificar el nivel de educación y formación que se necesita para utilizar el modelo con éxito. Incluye el nivel de sofisticación que se necesita para llevar a cabo el seguimiento del progreso, evitar riesgos inherentes y la pérdida de tiempo utilizando el modelo e identificar los beneficios que le llevan a utilizar el modelo en primer lugar.

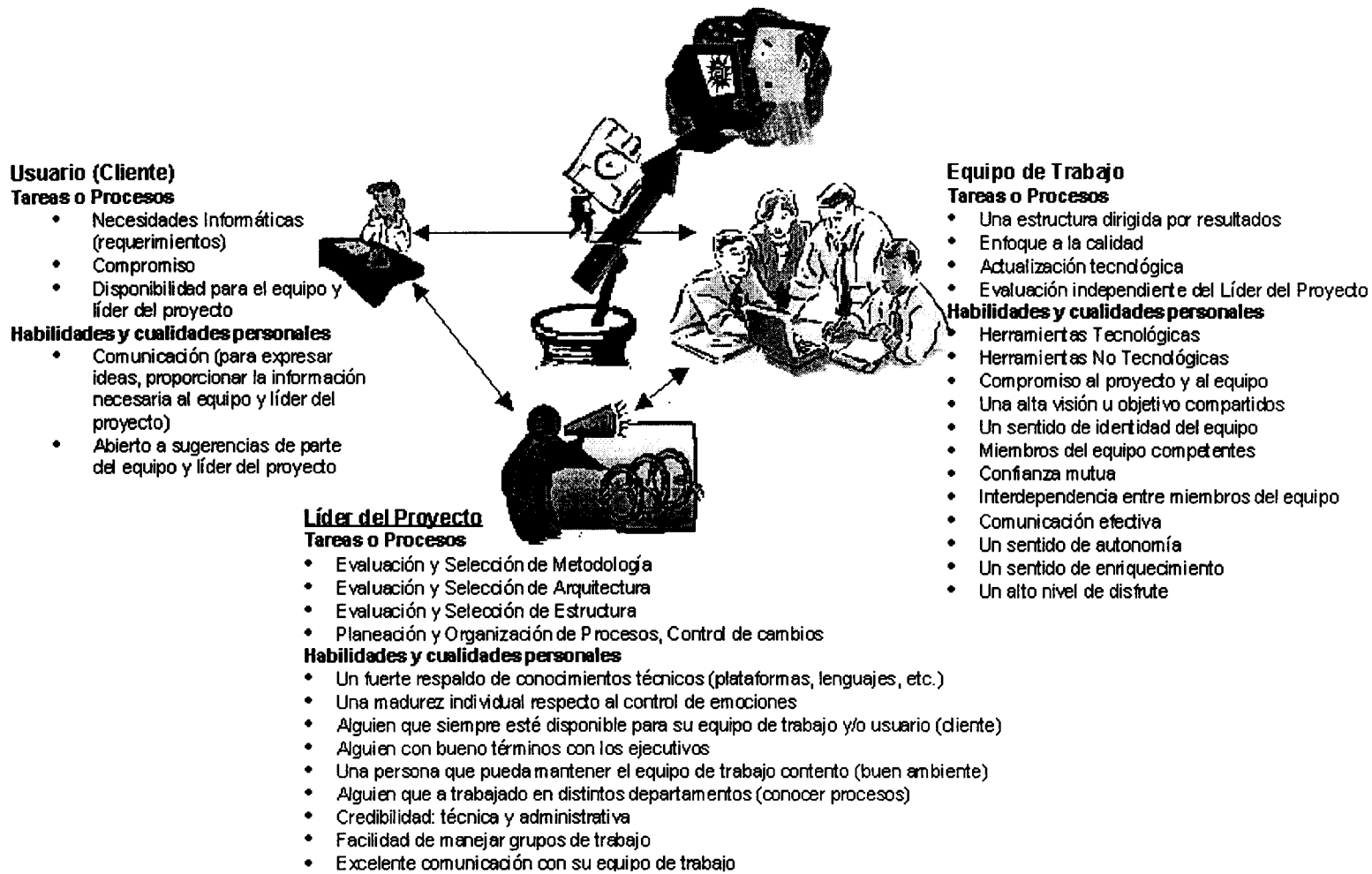
Anexo 5

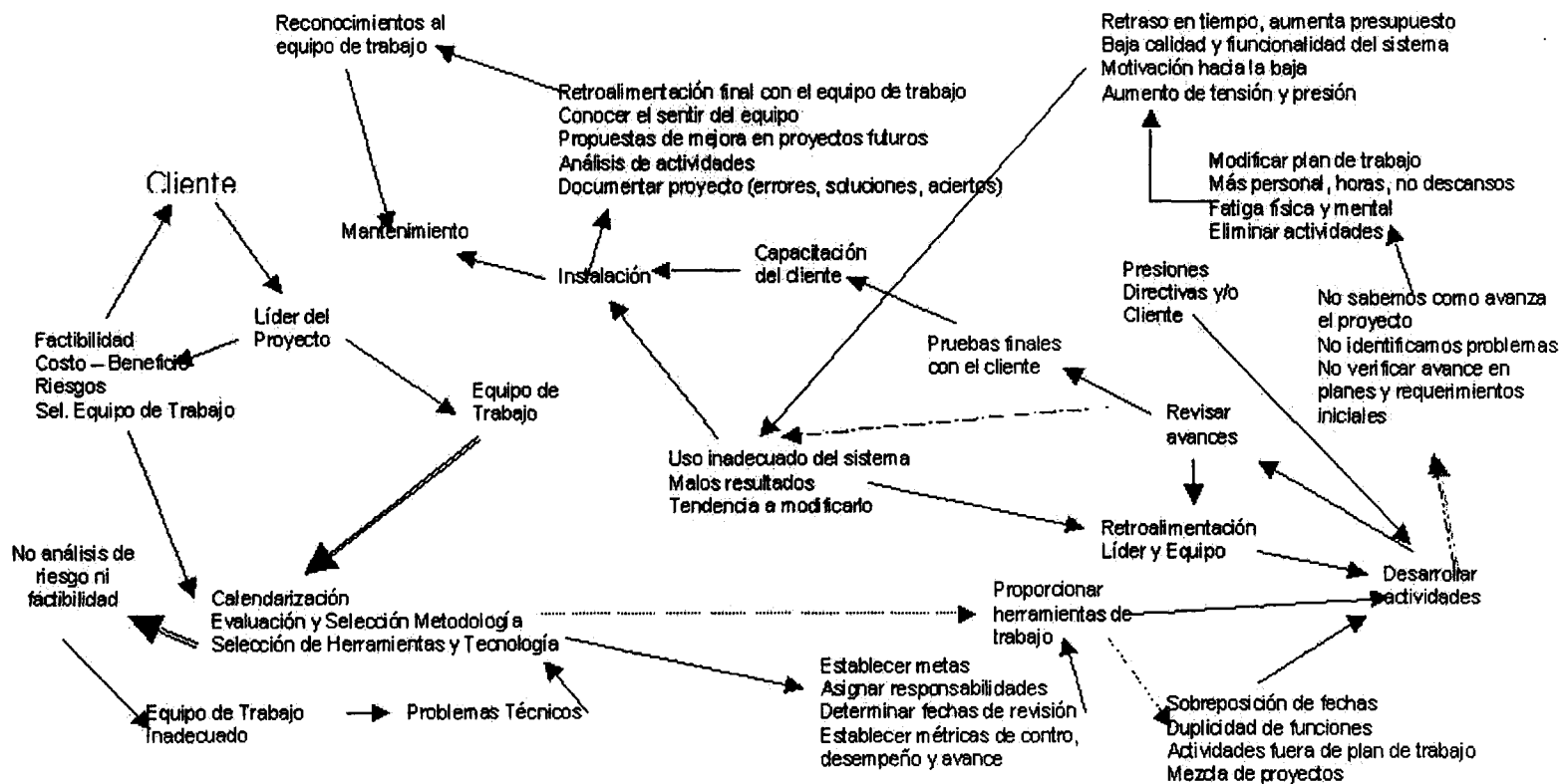
Problemas en el ambiente del proyecto



Anexo 6

Elementos de un proyecto exitoso





Mapa de errores y aciertos

Anexo 8

Tabla de Situaciones

Situación	Acciones	Riesgos de no tomarlo en cuenta
Recepción de solicitud de desarrollo	Contactar al usuario (cliente) para determinar los requerimientos exactos y no hacer suposiciones de lo que éste desea. Debe estar al menos un analista de sistemas.	Suponer que sistemas sabe lo que el usuario desea. El usuario no externa claramente sus necesidades, por lo tanto el desarrollo no estará de acuerdo a los requerimientos reales.
Análisis Preliminar	<p>Realizar un análisis preliminar sobre :</p> <ul style="list-style-type: none"> • Costo – Beneficio • Riesgos • Factibilidad <ul style="list-style-type: none"> ○ Económica ○ Humana ○ Tecnológica 	El no realizar un análisis previo, y lanzarse al análisis y diseño lógico, no es benéfico en la calidad del sistema, ya que no se determinó el equipo humano para desarrollar el proyecto, y no se determinó qué tecnología es la que se requiere para que el sistema se implante, esto provocará que no se tengan todos los elementos necesarios para llevarlo a cabo, ocasionando un incremento en el presupuesto.
Seleccionar al equipo de desarrollo	El Administrador del Proyecto debe tener el recurso humano capacitado para llevar a cabo el proyecto, en caso contrario, antes de comenzar a desarrollar, debe analizarse si se capacita al personal con que cuenta o bien se contrata a personal con los conocimientos y habilidades requeridas para llevar a cabo	El comenzar a desarrollar el sistema cuando no se tiene la capacitación y el conocimiento adecuado en el uso de alguna metodología de desarrollo, de un software de desarrollo o tecnología nueva, llevará al error de llevar a cabo el desarrollo

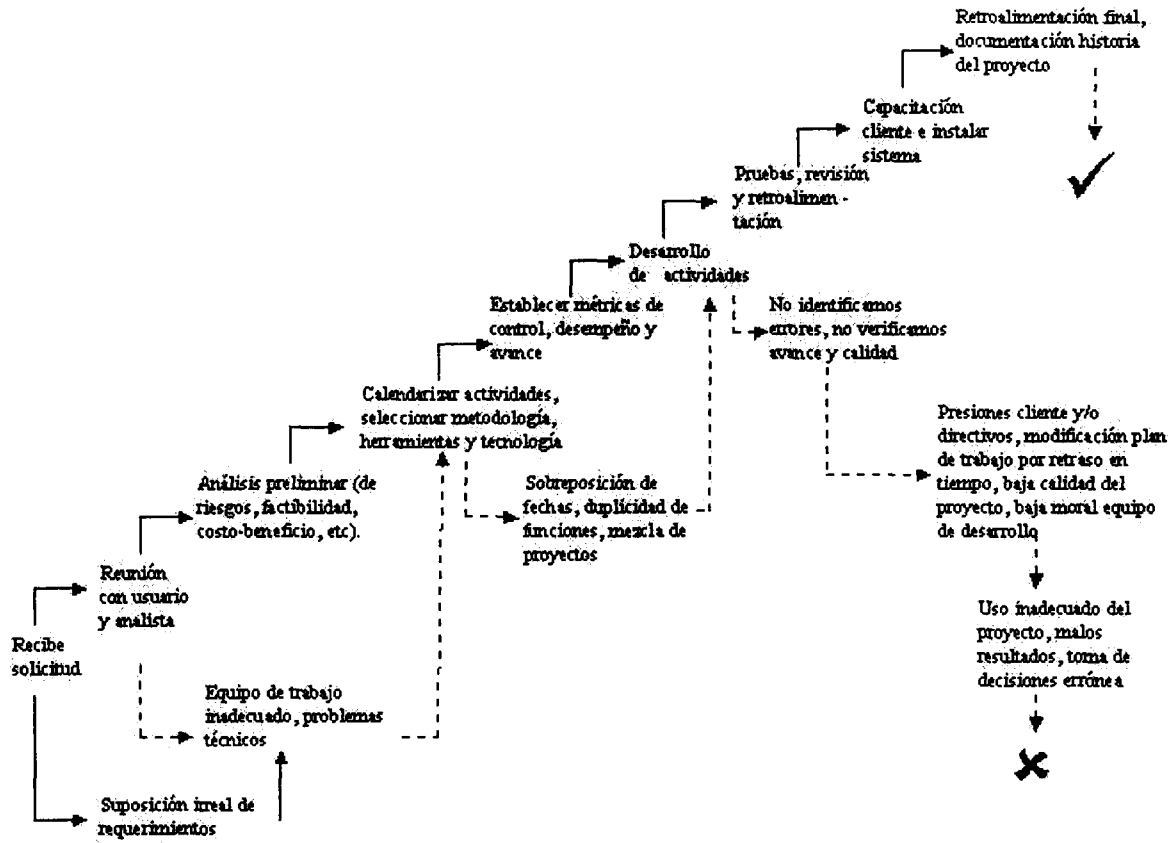
	<p>el proyecto, además debe tomar en cuenta la disponibilidad y carga de trabajo actual de cada uno de sus subordinados, evaluar las capacidades tecnológicas de cada uno. Debe seleccionar al personal que sea el más adecuado y que esté más disponible para llevar a cabo el proyecto.</p>	<p>“enfocado en la investigación” que es uno de los errores clásicos mencionados en el capítulo II. El no realizar un análisis de las capacidades y disponibilidad de cada uno de los elementos que colaborarán con el proyecto, provocará que se sobrecargue de trabajo a alguno de los elementos, o simplemente que se asignen las tareas no adecuadas al personal no adecuado. Invariablemente esto llevará un retraso en tiempo, incremento en costo, y un deficiente desarrollo encontrando diversos problemas técnicos, ocasionando un producto sin calidad.</p>
<p>Establecer metas, calendarización de actividades (incluye métricas de control, desempeño y avance, fechas de revisión)</p>	<p>Ayuda a determinar las actividades que deben estar desarrollándose, y de esta forma conocer en qué fase del proyecto nos encontramos. El establecer métricas, permitirá tener un control y una evaluación sobre el avance y calidad del proyecto, de esta forma se sabe cuando se debe de comenzar la siguiente etapa, cuando debe de terminar, y sobre todo tener con quien dirigirnos al momento de realizar una evaluación, ya que cada actividad tiene un responsable directo sobre su desarrollo.</p>	<p>Si no definimos como vamos a medir el avance y como se llevara a cabo el control de las actividades, simplemente no se llegará al resultado deseado. El no establecer responsabilidades en cada una de las actividades provocará el poco involucramiento del equipo de trabajo en el proyecto, ocasionando que los tiempos de cada actividad se incrementen, se cae en el error de tener</p>

		<p>duplicidad de funciones, y una pésima comunicación y colaboración entre los elementos que participan en el desarrollo del sistema.</p>
<p>Evaluación y selección de metodología de desarrollo</p>	<p>Antes de comenzar a desarrollar, primero debemos de saber cuál es la metodología de desarrollo que más se adecue al proyecto, la elección deberá ser responsabilidad del líder del proyecto, aunque puede compartirla con el equipo de desarrollo, el determinar una metodología requiere saber a fondo las características propias del proyecto, y obviamente, saber cuáles son las características de cada una de las metodologías, para hacer una comparación y de esta manera seleccionar la que más se acomode. El seleccionar una metodología no indica que no podamos tener una mezcla de las mismas, ya que bien podemos optar por un prototipo evolutivo, o simplemente mezclar el Code and Fix con la entrega por etapas por mencionar algún ejemplo.</p>	<p>Si no seleccionamos la metodología más adecuada a las características del proyecto, se caerá en el error de realizar actividades que no correspondan al proyecto, o simplemente que se mezclen actividades de diversos proyectos, también se reforzará el problema de la duplicidad de funciones y de fechas, provocando que no se tenga una adecuada coordinación de los miembros del equipo de desarrollo, provocando una incertidumbre sobre el avance del proyecto en el sentido de que no se puedan conectar las diferentes etapas por las cuales pasa el proyecto.</p>
<p>Selección de herramientas y tecnología de desarrollo</p>	<p>El contar con las herramientas y tecnologías necesarias, ayudará a que el desarrollo esté en perfectas condiciones y se cumplan con las fechas establecidas para cada una de las actividades.</p>	<p>El no contar con las herramientas y/o tecnologías de desarrollo que requiere el proyecto, simplemente ocasionará que no se tenga el sistema con los requerimientos</p>

		establecidos en el análisis preliminar y en el análisis lógico.
Desarrollo de actividades de acuerdo al plan establecido con la calendarización de actividades	Esto permite saber exactamente en que parte del proyecto nos encontramos, y nos dará la pauta para evaluar lo que se a hecho, así como saber qué es lo que sigue.	Si no se planean y calendarizan actividades, sencillamente emplearemos más tiempo del requerido para desarrollar el proyecto, provocando que el mismo se retrase o tenga actividades mal hechas, provocando que la calidad del mismo se vea afectada, debido a la falta de controles de avance.
Revisión del avance y retroalimentación	Permite evaluar el avance del proyecto, comparandolo con los requerimientos identificados, permitiendo conocer si existe algún desvío entre lo que se tiene y lo que se desea tenga el proyecto. Una retroalimentación adecuada en el tiempo adecuado, evitará que se modifiquen muchas actividades subsecuentes, y por consiguiente no se necesitará de más recursos para corregir los problemas que no se identificaron. Asegura de cierta forma la calidad del trabajo que se esta desarrollando.	El no contar con una revisión de avance y retroalimentación, además de no corregir los posibles errores que pueda tener el proyecto debido a que no los podemos identificar, provocará que el proyecto no se vaya a entregar con el tiempo y calidad requeridos, debido a que podemos confiar en que estará a tiempo cuando en realidad puede ser que esté atorado en algún punto. Esto irremediamente elevará el costo económico del proyecto, provocará que se tengan que realizar modificaciones a lo ya hecho, y pegará en la motivación del personal, aumentando así la presión que se pueda

		tener por no tenerlo a tiempo, provocando además que se eliminen actividades como las pruebas de calidad, por mencionar alguna.
Pruebas preliminares y/o finales	Es importante que se realicen estas pruebas por que es una manera de realizar una comparación de lo que se espera haga el sistema con lo que está realizando en la realidad. De esta forma podemos identificar los posibles desvíos que existen en el proyecto y se pueden tomar medidas para poder corregir y no volver a desviarse. Las pruebas finales nos darán la pauta para decidir si entregamos el proyecto o lo retrasamos debido a que se encuentren discrepancias en su funcionamiento.	El no realizarlas es arriesgarse a que el proyecto no sea exitoso, debido a que no sabemos cómo funciona en realidad, a pesar de que se confíe en la capacidad de su equipo de desarrollo. Si entregamos un producto si las pruebas requeridas para asegurar su calidad y funcionamiento, posiblemente el cliente no quede conforme y se niegue a pagar, o simplemente no lo volvamos a tener como tal.
Capacitación	Este proceso es de suma importancia para asegurar que el uso del sistema sea el adecuado.	El eliminar este problema, ocasionará que el usuario no sepa utilizar adecuadamente el sistema, y no obtener todos los beneficios que realmente le podría sacar de provecho al sistema.
Programa de mantenimiento	El contar con este tipo de programa, ayudará a garantizar que el uso y funcionamiento del sistema instalado sea el adecuado, y de esta manera se minimizar un impacto negativo en la organización al tener un mal uso del mismo.	El no contar con el, evitará que se detectaran posibles anomalías en el uso del sistema o bien en la integridad de la información manejada en el sistema.

<p>Reconocimientos</p>	<p>Los elementos del equipo de trabajo se sentirán motivados al saber que su trabajo es reconocido y recompensado en ocasiones en algunos sentidos, ya sea individual o en forma grupal, ya que es una muestra de hacerles notar que su trabajo es valioso y que ellos son parte fundamental del éxito de algún proyecto. Entre más alta sea la motivación que tenga un individuo para realizar su trabajo, este será realizado con una mejor calidad, sobre todo podemos contar con elementos bien identificados y que son leales a la empresa.</p>	<p>El no reconocer el esfuerzo de los elementos del equipo de trabajo, o simplemente el no hacerlos partícipes del éxito de un proyecto cuando este lo es, y que solamente se les identifique como culpables cuando algún proyecto no es exitoso si no todo lo contrario, afectará la tranquilidad y motivación de todos sus elementos, provocando que la calidad e intensidad de trabajo disminuya, inclusive se tendrá a personal descontento, que no realizará sus funciones como es debido, además de tener la posibilidad de perderlos en un futuro. El individuo para trabajar bien, necesita sentirse bien en el trabajo, y querer hacer las cosas por que sabe que es parte importante de su éxito.</p>
------------------------	--	---



Arbol de Decisiones

Anexo 9

Bibliografía

- [Ainsworth, 1999] Ainsworth Harold, "Why do projects break down?", Charter, Vol. 70, Febrero 1999, pags. 46-47
- [Alanís, 1998] Alanís Macedonio, "Definiendo las prioridades de la Dirección de Informática", Gobierno Digital, INFOLATINA, 29/02/2000
- [Bienkowski, 1989] Bienkowski Danek, "Ten causes of Project Busts", Computerworld, 1989, pag. 99
- [Biggs, 2000] Biggs, Maggie, "Technology won't end project failures; communication is key", Infoworld, Vol. 22, Enero 31 2000, pag. 70
- [Benjamin, 1978] Benjamin, Robert I., Control del ciclo de desarrollo de sistemas de información, LIMUSA, México, 1978
- [Curtis, 2000] http://www.cio.com/archive/101597_et_content.html, 27/06/2000
- [De la Torre, 1998] De la Torre, Ignacio, "Existe un software para que el gerente de sistemas controle sus recursos y proyectos de desarrollo", NET@, INFOLATINA, 08/05/1999
- [De León, 1997] De León, Elvira, Procedimiento que apoya el aprendizaje organizacional por medio de grupos de trabajo al establecer los elementos involucrados en el proceso de cambio de tecnología de información, Tesis de Maestría, ITESM, 1997
- [DeMarco y Lister, 1987] DeMarco, Tom., Lister, Timothy., Peopleware: Productive Projects and Teams, Dorset House, USA, 1987
- [Dorfman y Thayer, 1997] Merlin Dorfman y Richard H. Thayer foreword by Barry W. Boehm, Software Engineering , IEEE Computer Society Press, USA, 1997
- [Field, 1997] Field, Tom "When bad things happend to good projectts" www.cio.com/archieve/101597_bad_content.html, 01/02/99
- [Field, Pearson y Schuneider (2000)] www.cio.com/archive/120198_turk_content.html, 07/07/2000

- [Flores, 1993] Flores Magdalena, Sistema de Información para ejecutivos como una herramienta de apoyo para los negocios cuya fuerza impulsora es el servicio, Tesis de Maestría, ITESM, 1993
- [Gómez, 1992] Gómez Rosa, Guía para la introducción e implantación de sistemas estratégicos de información, Tesis de Maestría, ITESM, 1992
- [King, 1992] King David, Project Management made simple, Prentice Hall, New Jersey USA, 1992 pag. 3,
- [Liebowitz, 1999] Liebowitz, Jay, "Information systems: Success or failure?", The Journal of Computer Information Systems, Vol. 40, 1999, pag. 17
- [López, 1998] López Adán, Administración de Proyectos, ITESM, Monterrey México, 1998
Pag 2
- [Martínez, 1996] Martínez Cárdenas, Ignacio Alonso, Control de cambios en requerimientos de software durante las fases de desarrollo, Tesis de Maestría, ITESM, 1996
- [McConnell, 1997] McConnell, Desarrollo y Gestión de Proyectos Informáticos, McGraw Hill, España, 1997
- [McConnell, 1999a] <http://www.construx.com/bp05.html> 28/06/99
- [McConnell, 2000] <http://www.construx.com/stevemcc/rdenum.htm>,
06/07/2000
- [McConnell, 2000a] <http://www.construx.com/stevemcc/rdmistak.htm>,
06/07/2000
- [McConnell, 2000b] <http://www.construx.com/stevemcc/articles/art05.htm>,
06/07/2000
- [Meredith y Mantel, 1989] Mererith Jack y Mantel Jr. Samuel J., Project Management, Wiley, Second Edition, USA, 1989
- [Parsons y Oja, Parsons, June Jamrich, Oja Dan, New Perspectives on

- 1995] Computer Concepts, Course Technology, USA, 1995
- [Pereña, 1991] Pereña Brand, Jaime, Dirección y gestión de proyectos, Díaz de Santos, España, 1991
- [Radosevich, 1999] Radosevich Linda, " Measuring Up",
http://www.cio.com/archive/091599_project_content.html,
14/12/1999
- [Standish, 2000] Standish Group, "Unfinished voyages",
<http://standishgroup.com/visitor/voyages.htm>, 15/02/2000
- [Valdéz, 1997] Valdéz Jeaneth, Determinación de los factores clave para lograr una exitosa administración de proyectos de desarrollo de sistemas en empresas de consultoría, Tesis de Maestría, ITESM, 1997
- [Weiss y Wysocki, 1992] Weiss Joseph W. y Wysocki Robert K., 5-Phase Project management, Perseus Books, USA, 1992
- [Yourdon, 1988] Yourdon Edward, Managing the system life cycle, Prentice Hall, USA, 1988

Centro de Información-Biblioteca



30002005919428