



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**



**FACULTAD DE ESTUDIOS PROFESIONALES  
ACATLAN**

**LA ADMINISTRACIÓN DE PROYECTOS EN EL  
DESARROLLO DE SOFTWARE.**



**SEMINARIO TALLER  
EXTRA CURRICULAR  
QUE PARA OBTENER EL TÍTULO DE:  
LICENCIADO EN MATEMÁTICAS  
APLICADAS Y COMPUTACIÓN  
P R E S E N T A :  
JOSÉ LUIS VALVERDE ELIZALDE**

**ASESOR:  
LIC. JUAN TORRES LOVERA**

**NOVIEMBRE DE 2004**



Universidad Nacional  
Autónoma de México



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

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

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

*A mis padres y hermanos que siempre  
me han regalado su cariño y apoyo.*

*Para ti papa por ejemplo de la tenacidad en el trabajo.*

*Para ti mama por la bondad y el apoyo de toda la vida.*

*A Marisela porque siempre es mi incondicional de apoyo.*

*A Edgar porque siempre me ha brindado su apoyo.*

*Agradezco a todos mis compañeros del seminario, por su apoyo para continuar con el proyecto que iniciamos.*

*Adriana, Elizabeth, Karla, Lupita, Federico, Gina, Mariel, Mary, Martha, Rosario, Patricia, Raquel, Roció, Soco, Thelma, Valentin y Viridiana,*

*A mis profesores de seminario:*

*Luz Maria Lavin Alanis  
Hugo reyes Martinez  
Ignacio Martín Lizárraga Gaudry  
Juan Miguel Ramirez Zozaya  
JuanTorres Lovera*

*Agradezco a mi asesor Juan Torres Lovera por el apoyo brindado al presente trabajo, por su orientación y sus observaciones para logran la conclusión de este, gracias.*



## **CONTENIDO**

<b>INTRODUCCIÓN.....</b>	<b>iv</b>
<b>CAPITULO 1. ....</b>	<b>1</b>
<b>PROYECTOS DE DESARROLLO DE SOFTWARE Y SU ADMINISTRACIÓN. ....</b>	<b>1</b>
1.1 ADMINISTRACIÓN.....	3
1.1.1. FUNCIONES DE LA ADMINISTRACIÓN. ....	4
1.2 ADMINISTRACIÓN DE PROYECTOS. ....	5
1.2.1. GRUPO.....	5
1.2.2. EQUIPOS.....	6
1.2.3. DIFERENTES TIPOS DE PAPELES.....	6
1.3 TIPOS DE ESTRUCTURA ORGANIZACIONAL. ....	7
1.3.1. ESTRUCTURA ORGANIZACIONAL POR EQUIPOS.....	7
1.3.2. ESTRUCTURA ORGANIZACIONAL TRADICIONAL.....	7
1.3.3. ESTRUCTURA ORGANIZACIONAL POR PROYECTOS.....	8
1.4 EL SOFTWARE.....	9
1.4.1. DEFINICIÓN DE SOFTWARE.....	9
1.4.2. EVOLUCIÓN DEL SOFTWARE.....	10
1.4.3. ERAS DEL SOFTWARE.....	11
1.4.4. DESARROLLO DE SOFTWARE.....	14
1.5 LA IMPORTANCIA DE ADMINISTRAR LOS PROYECTOS DE SOFTWARE.....	16
1.6 PROBLEMÁTICA EN EL DESARROLLO DE PROYECTOS DE SOFTWARE.....	18
1.6.1. DEFINICIÓN DE VARIABLES DE LA PROBLEMÁTICA.....	20
1.7 CONCLUSIONES.....	24
FUENTES DE CONSULTA.....	26
<b>CAPITULO 2. ....</b>	<b>27</b>
<b>LA PLANEACION DEL DISEÑO DE PROYECTOS DE SOFTWARE.....</b>	<b>27</b>
2.1 PREPARACION DEL PROYECTO.....	29
2.2 DEFINICION DEL PROYECTO.....	31
2.3 ORGANIZACIÓN.....	32
2.3.1 ATRIBUTOS DE LAS PERSONAS PARA FORMAR EQUIPOS.....	32
2.3.2 EL NÚMERO DE INTEGRANTES PARA FORMAR UN EQUIPO.....	35
2.3.3 LAS PERSONAS INVOLUCRADAS EN EL PROYECTO.....	36
2.3.4 ASIGNACION DE RECURSO HUMANO AL PROYECTO.....	37
2.4 COMUNICACIÓN EN EL PROYECTOS.....	38
2.5 LOS RECURSOS MATERIALES Y COSTOS DEL PROYECTO.....	38
2.5.1 COSTOS DEL PROYECTO.....	39
2.6 DOCUMENTACIÓN DEL PROYECTO.....	40
2.6.1 PLAN DE ADMINISTRACION DEL PROYECTO CON EL ESTANDAR IEEE.....	41
2.7 CONCLUSIONES.....	43



FUENTES DE CONSULTA .....	45
<b>CAPITULO 3.</b> ....	<b>46</b>
<b>ELEMENTOS PROSPECTIVOS DEL DISEÑO.</b> .....	<b>46</b>
3.1 <b>EL MODELO DE CAPACIDAD DE MADUREZ (CMM)</b> .....	<b>47</b>
3.2 <b>ESTANDAR ISO 12207.</b> .....	<b>50</b>
3.3 <b>PROCESO ADMINISTRATIVO PROPUESTO PARA EL DESARROLLO DE SOFTWARE.</b> .....	<b>53</b>
3.3.1    DEFINICION DEL PROYECTO. ....	53
3.3.2    PLANEACIÓN DEL PROYECTO.....	54
3.3.3    PROGRAMACION Y CONTROL DE PROYECTO.....	55
3.3.4    ASIGNACION DE RESPONSABILIDADES.....	57
3.3.5    ENTORNO DE TRABAJO.....	58
3.3.6    DIRECCION .....	59
3.4 <b>ESCENARIOS.</b> .....	<b>61</b>
3.4.1    VARIABLES PRINCIPALES: .....	63
3.4.2    VARIABLES SECUNDARIAS:.....	64
3.4.3    FORMULACIÓN DE ESCENARIOS.....	65
3.5 <b>CONCLUSIONES</b> .....	<b>69</b>
FUENTES DE CONSULTA .....	71
<b>CONCLUSIONES GENERALES</b> .....	<b>72</b>
<b>BIBLIOGRAFIA</b> .....	<b>75</b>



## **TITULO**

LA ADMINISTRACIÓN DE PROYECTOS EN EL DESARROLLO DE SOFTWARE

## **INTRODUCCIÓN.**

La velocidad con la que se dan los cambios hoy día, nos obligan a estar en un continuo aprendizaje, que permita afrontar mejor cada problemática que se presenta. Uno de estos cambios ha sido el nivel tecnológico que tenemos hoy día, esto se da en las comunicaciones, en el deporte, o en la investigación. Estos cambios algunas veces llegan a pasar inadvertidos, pero uno de los que mayor impacto ha tenido, es la aparición de las computadoras, con la que se han realizado muchas actividades que de alguna manera afectan a nuestras vidas, positivamente o negativamente. Pero el funcionamiento de las computadoras no sería posible, si esta no tuviese una forma de interactuar con el hombre, esta interacción se logra con base a creación de software, actividad de apenas cinco décadas de realizarse a gran escala. Es tan importante el desarrollo de software para computadora que hoy en día es parte de un gran mercado mundial, esta importancia hace que desde el inicio del desarrollo hasta su finalización, encierra un gran número de tareas, las que deben realizarse hasta que este es puesto en la computadora. Estas tareas involucran requerimientos previamente establecidos para la construcción de software, las que son realizadas en diferentes etapas o actividades para su creación, las que están basadas en un proceso administrativo, el que debe asegurar la finalización del proyecto emprendido.

El presente trabajo tiene como finalidad el proponer un diseño administración de desarrollo de software, con el objetivo principal de Administrar proyectos de software para la optimización de recursos, que aseguren el cumplimiento de los objetivos para los que se plantea un proyecto. Por la importancia que tiene el software, desde el desarrollo hasta que es colocado en una computadora para su utilización, las tareas necesarias para este proceso de construcción de software tienen que ser administradas de una manera eficaz para la creación de software, para realizar una administración, resulta importante comprender las funciones que la administración realiza, por lo que en el primer capítulo del presente trabajo se hace una serie de definiciones de administración y de sus principales funciones, con la idea de entender como la administración ayuda a desarrollar proyectos de



software. En este mismo capítulo al hablar de software se define su concepto, las características que tiene, su ciclo de desarrollo, además se hace una semblanza de la evolución que han tenido los sistemas y la complejidad que ha generado al desarrollar software que cada vez se vuelve más robusto en las diferentes etapas por las que a pasado, se describen las principales causas de la problemática que tiene el desarrollo de software, que no son problemas únicos de una organización, sino que es parte de la complejidad que han creado en los desarrollos de software. Se determina la necesidad de administrar los proyectos de software y las causas de la problemática por la que pasa, realizando un diagnóstico de la problemática, para que al formular la relación de variables con la problemática planteada se establecen posibles soluciones a la problemática, proponiendo posibles alternativas de solución.

Habiendo realizado un análisis de las principales causas que provocan la problemática de proyectos de software, es decir después de un diagnóstico, se plantean en el capítulo dos, las tareas a realizar para desarrollar un diseño de proyecto ideal, que permita dar seguimiento a los proyectos nuevos. Diseño que determinan las etapas que tendrá que considerarse para la realización de proyectos, considerando la formación de equipos de trabajo como forma de organización del proyecto, se establecen una serie de atributos necesarios de considerar para el personal que forma parte de un equipo, el número de integrantes, se determina los costos a considerar y se menciona la comunicación que puede establecerse en un equipo de trabajo para tener un desarrollo organizacional adecuado, que facilite el desempeño de las actividades del proyecto, se propone el plan de documentación de proyectos de software del Instituto de Ingenieros electrónicos y eléctricos (IEEE Institute of Electronic and Electrical Engineers), como norma a seguir para la documentación de proyectos, ya que la documentación es una de las partes importantes de un proyecto de software.

Para el capítulo tres se hace referencia a dos estándares internacionales de procesos de desarrollo de software, el CMM (Modelo de Capacidad de Madurez) y el estándar ISO (Organización Internacional de Estándares) con la norma 12207, que norman los procesos de desarrollo de software, con estos estándares presentados como marco de referencia. Se realiza la propuesta del proceso administrativo a seguir en el desarrollo de software, en el que se determina el proceso administrativo, considerando los planteamientos realizados en el capítulo dos como punto de partida para la realización de este proceso administrativo. El diseño propuesto determina las tareas a seguir, como es la definición de



---

proyecto, la planeación como función administrativa, el programa de actividades y el control, la importancia que tienen las graficas de Gantt para supervisar y dar seguimiento a proyectos. Se establecen las responsabilidades y se realiza la asignación de actividades para cada persona, se menciona el entorno de trabajo propicio para el desarrollo del proyecto, que cree un ambiente de trabajo idóneo para el personal. Se planean los elementos de dirección y se finaliza el capitulo con el planteamiento de escenarios prospectivos, que son determinados para periodos de tiempo de cinco, diez y quince años, en el que se establecen los posibles escenarios en los que los proyectos de software serán realizados para un futuro próximo. En el que la capacitación, la tecnología serán una de las principales formas de lograr realizar proyectos y lograr estar en un nivel de desarrollo profesional que permita en un corto plazo una certificación en este tipo de desarrollo profesional.



## CAPITULO 1.

### PROYECTOS DE DESARROLLO DE SOFTWARE Y SU ADMINISTRACIÓN.

Objetivo. Definir los conceptos de administración, así como los elementos del desarrollo de software, para poder establecer un diseño, que permita la solución de las principales debilidades que se presentan en el desarrollo de software, en la que se establezcan los elementos mínimos necesarios para la realización de cada proyecto.

A finales de las últimas tres décadas, la computadora ha sido y es una herramienta de trabajo, que ha ayudado a desarrollar actividades de todo tipo, ya sea de negocios, de investigación, académicos y de diferentes áreas de conocimiento. Pero el funcionamiento de la computadora no sería posible, si esta no tiene los elementos necesarios para su utilización. Uno de estos, es el software que hace posible que esta pueda tener una comunicación con las personas, para poder ponerla en funcionamiento. En otras palabras el software es el sistema que la computadora utiliza para poder hacer posible la comunicación con las personas y/o con otras computadoras. Hoy día el desarrollo de software es una de las actividades que más crecimiento ha tenido, tanto comercial como en investigación. Al ser un producto necesario e importante para muchas organizaciones, las mismas que crean áreas dedicadas a desarrollar software para cubrir sus propias necesidades, que posteriormente si el producto realizado es exitoso pueden incluso llegar a ser colocados en el mercado. El software es hoy una innovación tecnológica, que permite realizar actividades que de otra forma serían más complejas o más difíciles de realizar, permite agilizar procesamiento de datos para cubrir necesidades de información, de investigación, de diversión, de hacer negocios, ya que hoy día estamos en lo que se ha dado por llamar la **revolución informática**.<sup>1</sup>

Con la aparición de los procesadores de palabras y la hoja de cálculo, se da inicio a los desarrollos de software a gran escala, desarrollos que cada vez requieren de mayor número de recursos, tanto económicos, humanos y técnicos. Este tipo de proyectos establecieron las primeras formas de realizar proyectos de desarrollo de software.

---

<sup>1</sup> "Historia de la computación". [www.monografias.com](http://www.monografias.com)



Cuando se desarrollaron los primeros programas de software, los sistemas no eran de grandes dimensiones y era posible realizarlos con pocas personas, pues quienes realizaban el desarrollo de la programación, eran los mismos que se encargaban de realizar las pruebas, validar resultados e implantar sus desarrollos. Cuando los proyectos crecieron en requerimientos, como en el manejo de equipos de cómputo cada vez más sofisticado, las actividades de cada proyecto se hicieron cada vez más complejas, por lo que se recurrió a especializar cada actividad que era requerida en cada nuevo desarrollo. Así los proyectos empezaron a crecer y se dio inicio a lo que ha dado en llamarse **una crisis del software**, debido a que el desarrollo se convirtió cada vez más en una tarea compleja, por los elementos necesarios para la construcción de este.

Hoy día muchos proyectos tienen lugar en diferentes organizaciones, debido a que realizan su propio software o desarrollan software para ponerlo en el mercado, pero la tarea de desarrollar no ha sido fácil, ya que ha afrontado cambios constantes dentro de las organizaciones, los niveles de competitividad, el rápido crecimiento tecnológico, que provocan que los proyectos lleguen a tener problemas en su planeación, provocando problemas en su desarrollo, en su finalización y que no cumplen con el requerimiento para el que se ha realizado.

Un estudio realizado en los Estados Unidos en 1998 por la compañía Standish Group, reveló de un análisis de más de 28 mil proyectos de desarrollo de sistemas<sup>2</sup>, que un 28 por ciento los proyectos habían fracasado rotundamente (suspendidos o cancelados antes de terminar) y otro 46 por ciento habían terminado fuera de tiempo, arriba de presupuesto o con menor alcance del planeado originalmente y solo el 26 por ciento habían logrado terminar con los términos previamente establecidos del proyecto.<sup>3</sup> *Esto refleja el grado de éxito que puede tenerse en el desarrollo de software, y el porqué de la necesidad de realizar una administración eficiente, que permita solucionar el retardo de la entrega de proyectos de software en un tiempo determinado y con un presupuesto establecido.*

Los desarrollos de software requieren cada vez más de mayores recursos, nuevas metodologías de desarrollo, una constante actualización, una mayor atención por parte de

---

<sup>2</sup> Sistemas. Para esta investigación, utilizare el término de sistema como el producto del desarrollo final de un software.

<sup>3</sup> Roberto Toledo. Administre mejor sus proyectos: 12 pasos básicos para el éxito. [www.alpha-hardin.com](http://www.alpha-hardin.com). Artículo con fecha 2004-03-01



quienes están involucradas en esta actividad profesional, para desarrollar alternativas de solución, donde se realice una administración cada vez más eficiencia en cada proyecto que es emprendido. Hoy día la administración de proyectos se ha convertido en una herramienta para muchas organizaciones ya que el medio ambiente en el que hoy se realizan proyectos es complejo y cambia con rapidez, la tecnología es incierta y compleja sin no se le sabe manejar y el nivel de competitividad entre las organizaciones es cada vez mayor.

El presente trabajo de investigación, propone realizar un diseño permita establecer alternativas para administrar proyectos de desarrollo de software. Así con este concepto se hace una serie de definiciones de administración, definiciones que son utilizados en el presente trabajo, para poder desarrollar un modelo ideal de administración. Así también se definen los conceptos referentes al software, para poder establecer el desarrollo tanto del software como de cada proyecto que será administrado.

Por lo que se definen los principales conceptos que son utilizados en la administración, para establecer un marco referencial de la administración de proyectos de software.

## **1.1 ADMINISTRACIÓN<sup>4</sup>**

### **Definición:**

Administración se refiere al proceso de realizar, coordinar e integrar las actividades de trabajo para que estas se lleven a cabo en forma eficiente y eficaz con otras personas y por medio de ellas.

El proceso de la administración representa las funciones o actividades primarias con las que los administradores están comprometidos, estas funciones son tradicionalmente clasificadas como planeación, organización, dirección y control.

---

<sup>4</sup> Stephe Robbins, David A de Cenzo, "Fundamentos de la administración", conceptos y aplicaciones, México, Pag. 5



### 1.1.1. FUNCIONES DE LA ADMINISTRACIÓN.<sup>5</sup>

**Planeación.** La planeación define las metas de una organización, establece una estrategia general para alcanzar esas metas y desarrolla una jerarquía general de los planes para integrar y coordinar las actividades.

**Organización.** Es la función de diseñar las actividades que se deben de realizar, quien las realizara, como deben de ser agrupadas las actividades, quien reporta a quien, y que decisiones deben de ser tomadas.

**Dirección.** Es la forma de dirigir a las personas dentro de la organización, es también la encargada de motivar a sus empleados, dirigir las actividades de otras personas, seleccionan el canal de comunicación más efectivo, encargado de resolver conflictos entre los miembros.

**Control.** Es el proceso de comparar y corregir el desempeño de la organización. El desempeño real tiene que ser comparado con las metas establecidas anteriormente por la misma organización.

**Proyecto.** Un proyecto es un conjunto de actividades que se realizan una sola vez y cuyos puntos de inicio y finalización ya se han definido.<sup>6</sup>

**Proyecto de software.** *Un proyecto de software es el conjunto de actividades necesarias para producir los artefactos requeridos. Incluye contacto con el cliente, escribir la documentación, desarrollar el diseño, escribir el código y probar el producto.*

De las definiciones establecidas para la administración, debemos también definir qué es la administración de proyectos, para poder plantear como la administración de proyectos puede utilizarse para gestionar proyectos de desarrollo de software.

---

<sup>5</sup> Ibid Pag. 6.

<sup>6</sup> Robbins, Coulter, Administración. Prentice Hall. México. Pag. 13



## **1.2 ADMINISTRACIÓN DE PROYECTOS.**

La administración de proyectos es una tarea que consiste en asegurarse de que las actividades se realicen a tiempo, dentro del presupuesto y de acuerdo a las especificaciones establecidas.<sup>7</sup>

La administración de proyectos está sujeta a actividades técnicas complejas y de cambio, ya que los productos de software que se desarrollan, requieren de participantes con diversas habilidades y conocimientos. Los mercados competitivos y los requerimientos evolutivos introducen cambios en el desarrollo, desencadenan frecuentes reasignaciones de recursos y dificultan el seguimiento del proyecto. Los modelos de administración permiten representar los recursos disponibles para el proyecto, las restricciones a las que están sujetos los recursos y las relaciones que deben tener entre estos.<sup>8</sup>

Los proyectos ya sean grandes o de pequeño alcance organizacional, deben tener una estructura organizacional, y esta puede establecerse con la creación de grupos o equipos de trabajo, para que puedan ser integrados en la elaboración de un proyecto. Al ser esta una de las formas de organización, se define el concepto de grupo y la de equipo de trabajo que son utilizadas en la administración de proyectos.

### **1.2.1. GRUPO.**

Un grupo se define como dos o más individuos interdependientes que interactúan, se reúnen para alcanzar objetivos particulares. Los grupos pueden ser formales o informales. Los grupos formales son grupos de trabajo establecidos por las organizaciones que tienen asignados trabajos y actividades establecidas. En los grupos formales, los comportamientos que se deben tener son estipulados y dirigidos hacia las metas organizacionales. En contraste los grupos informales son de naturaleza social. Estos grupos son formaciones naturales que aparecen en el entorno de trabajo como respuesta a la necesidad de contacto social.<sup>9</sup>

---

<sup>7</sup> Ibid Pag. 289

<sup>8</sup> Bruegge Bernd, Dutoit Allen H., "Ingeniería de software orientada a objetos", Prentice Hall, México. 2002. Pag. 407-410

<sup>9</sup> Robbins, De Censo. Fundamentos de Administración. Prentice Hall, Mexico, 1999. Pag. 266



### **1.2.2. EQUIPOS.**

La estructura de un equipo de trabajo es una estructura temporal creada para cumplir una actividad específica, bien definida que requiere de la participación personal de un número de subunidades de la organización. Puede ser considerada como una versión a menor escala de la matriz organizacional temporal. Los integrantes prestan sus servicios en el equipo hasta alcanzar su objetivo. Luego el equipo de trabajo es disuelto y sus miembros son asignados a un nuevo equipo, o regresan a su departamento funcional o dejan la organización.

#### **Los equipos de trabajo.**

Los equipos se convierten cada vez más en medios importantes alrededor de los cuales se asigna trabajo. Ya que los equipos generalmente tienen un mejor desempeño que los individuos cuando las actividades realizadas requieren de habilidades múltiples, juicio y experiencia. A medida que las organizaciones se reestructuran para competir con mayor eficiencia y efectividad, se vuelven hacia los equipos como una forma de utilizar mejor los talentos de sus empleados. Los equipos son más flexibles y por tanto dan mejor respuesta a un entorno cambiante en las agrupaciones burocráticas de empleados. También pueden ser reunidos con mayor rapidez, disgregarse, cambiar su énfasis y eliminarlos por completo. Estos son grupos formales, integrados por individuos interdependientes, responsables de alcanzar una meta.

### **1.2.3. DIFERENTES TIPOS DE PAPELES.**

Un rol se le define a una serie de patrones de comportamiento esperado y atribuible a alguien que ocupa cierta posición en una unidad social. Los individuos desempeñan papeles múltiples, ajustándolos al grupo o equipo al que pertenecen en ese momento. En una organización los empleados intentan determinar qué comportamiento se espera de ellos.<sup>10</sup>

---

<sup>10</sup> Robbins Stephen P., Coulter Mary, Administración, Prentice Hall, Mexico, 2000. Pag. 35.



Dentro de toda organización, siempre existe una forma de determinar la forma en que se organizan las personas pertenecientes a dichas organizaciones. Existen varias de las que se puede establecer varios tipos de estructura organizacional, se mencionan tres tipos de estructuras utilizadas en una organización: la tradicional, por proyectos y por equipos.

### **1.3 TIPOS DE ESTRUCTURA ORGANIZACIONAL.**

Una organización es un conjunto formal de personas y otros recursos establecidos en función del cumplimiento de un conjunto de metas. Y una estructura organizacional se refiere a las subunidades organizacionales y a la relación de estas con la organización general, dependiendo de las metas de la organización y de su método administrativo, es posible utilizar diversas estructuras. Estas estructuras pueden ejercer un importante impacto en la concepción de los sistemas de información y en el tipo que de estas estructuras sean empleadas. Aunque a este respecto existen posibilidades de diseñar una propia, una estructura organizacional suele corresponder a una de las siguientes categorías: *tradicional, de proyectos, de equipos o multidimensional*.<sup>11</sup>

#### **1.3.1. ESTRUCTURA ORGANIZACIONAL POR EQUIPOS.**

Estructura formada por equipos durante la elaboración del trabajo o grupos de trabajo. En algunos casos los equipos son pequeños; en otros pueden ser muy grandes, por lo común cada equipo cuenta con un líder, el cual rinde informes a un administrador de alto nivel en la organización. Dependiendo de las tareas que se realizaran, el equipo que se conforme puede ser provisional o permanente.

#### **1.3.2. ESTRUCTURA ORGANIZACIONAL TRADICIONAL.**

En el marco del tipo de estructura conocida como estructura organizacional tradicional, directores de departamentos responden por lo general a criterios de división acordes con sus funciones, pudiendo ser mercadotecnia, producción, sistemas de información, finanzas, contabilidad, investigación o desarrollo, etc. A la estructura organizacional tradicional también se le conoce como *estructura jerárquica*, pues es posible concebirse como una serie de niveles, cuyas capas más altas disponen de mayor poder y autoridad en la organización. En la actualidad se ha reducido la cantidad de niveles, o capas, administrativos de la estructura organizacional tradicional. (Figura. 1.1)

<sup>11</sup> Raip M.Stair. Principios de sistemas de información. Mexico Pag. 45



### Estructura organizacional tradicional o por funciones.

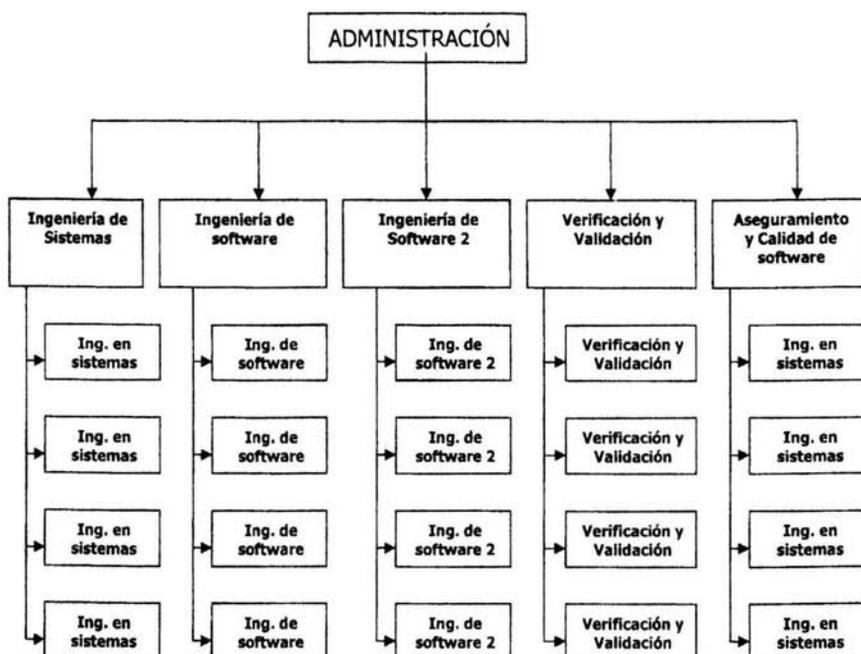


Figura 1.1

### 1.3.3. ESTRUCTURA ORGANIZACIONAL POR PROYECTOS.

Una estructura organizacional por proyectos se centra en grandes productos o servicios. Muchos equipos de proyectos son de carácter tradicional. Una vez concluido el proyecto, los miembros del equipo responsables se integran a nuevos equipos, encargados a su vez de realizar otros proyectos. (Figura 1.2)



## Estructura organizacional por proyectos.<sup>12</sup>

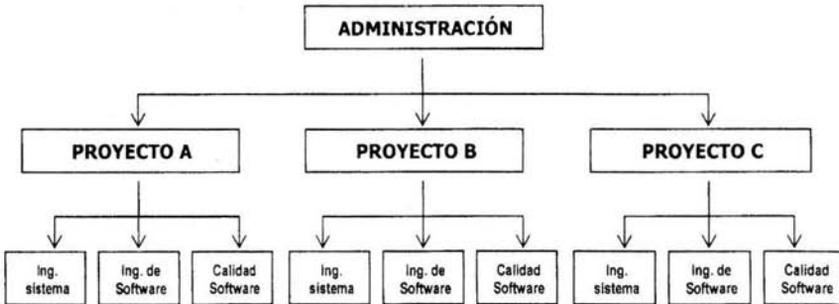


Figura 1.2.

### 1.4 EL SOFTWARE

Para poder realizar el diseño de un proyecto de desarrollo de software, debemos comprender que es el software, sus características, la evolución que ha tenido, las dificultades que se han presentado para su creación, para tener un referente de cómo poder entenderlo. Para ello se define al software y se realiza una reseña histórica de su evolución hasta nuestros días.

#### 1.4.1. DEFINICIÓN DE SOFTWARE.

El software consiste de programas de computación que controlan las operaciones del hardware de un equipo de cómputo, dichos programas computacionales consisten en sucesiones de instrucciones para la computadora, haciendo posible su funcionamiento, para procesar datos y proporcionarnos información.

El software es la parte lógica que permite operar una computadora, tiene características que lo hacen diferente a la parte física que compone una computadora, estas características son las siguientes:

<sup>12</sup> Stair, Reynolds. Principios de sistema de información. Thomson. 4ta edición. México 2000. Pag. 47



### **Características del software.<sup>13</sup>**

- El software se desarrolla no se fabrica.
- El software no se estropea.
- El software es creado a la medida.
- El software tienen una vigencia de vida, en relación directa con el desarrollo de tecnológico del hardware.

Estas características hacen que el software sea una parte no fácilmente manejable, y que su desarrollo deba considerar todos los aspectos que involucran un desarrollo de este tipo.

Durante las últimas tres décadas el software se ha tenido un desarrollo de varias etapas de tiempo, las que se han identificado como eras del software, estas han ayudado a identificar su evolución del mismo, entender su evolución nos proporciona elementos para poder plantear desarrollos, de acuerdo a la complejidad por la que ha pasado, por lo que se realiza una breve semblanza de las etapas por las que ha pasado el software.

### **1.4.2. EVOLUCIÓN DEL SOFTWARE.**

A principios de los años 80 el software todavía era desconocido para la mayoría de las personas, las compañías de software dedicadas al software comercial, como Microsoft, no existían, y en menos de dos décadas se desarrollaron tecnologías basadas en software, que rápidamente han cambiando.

El software de computadora, está inmerso en sistemas de todo tipo, ya sean de transporte, sistemas de diagnósticos médicos, de telecomunicaciones, de desarrollos militares, procesos industriales, entretenimiento, productos de oficina, etc., todo esto ha hecho que cambie la percepción que se tiene del software.

Hoy día el software reside dentro de un teléfono celular u opera dentro de una computadora central, el software es un transformador de información, produciendo, gestionando, adquiriendo, modificando, mostrando o transmitiendo información que puede ser tan simple como un solo bit, o tan complejo como una simulación en multimedia. Como vehículo utilizado para hacer entrega del producto, el software activa la base de control de la

---

<sup>13</sup> Pressman Rogger, Ingeniería de software. McGraw hill. México, 1999. Pag 8.



computadora (sistemas operativos), la comunicación de información (redes), y la creación y control de otros programas (herramientas de software y entornos). El software gestiona la entrega de información, transforma datos en información, para que estos sean útiles en un contexto local; gestiona información comercial para mejorar la competitividad; proporciona el acceso a redes de información por todo el mundo (por ejemplo: Internet); y también proporciona el medio de adquirir información en todas sus formas.

El papel del software informático ha sufrido un cambio significativo en la segunda mitad del siglo veinte. Enormes mejoras en rendimiento del hardware, profundos cambios de arquitecturas informáticas, grandes aumentos de memoria y capacidad de almacenamiento, han permitido la creación de sistemas más sofisticados y más complejos basados en computadora.

Durante los años 70 y 80 se pronosticaba que las computadoras y el software habrían de dar un gran impacto cultural. Esto a dado en llamarse una nueva *revolución industrial*, donde la humanidad habría de convertirse en una sociedad de la información, se predijo que la información y el conocimiento serían el foco de poder del siglo veintiuno.<sup>14</sup>

### **1.4.3. ERAS DEL SOFTWARE.**

#### **La primera era del software.**

Durante los primeros años de la era de la computadora, el software se contemplaba como un complemento al hardware. La programación de computadoras era un arte, para el que existían pocos métodos sistemáticos. El desarrollo del software se realizaba virtualmente sin ninguna planificación, hasta que los planes eran difíciles de controlar y los costos empezaron a aumentar. Los programadores trataban de hacer las cosas bien, y con un esfuerzo heroico, a menudo salían con éxito.

Durante los primeros años de desarrollo de software, es decir de finales de los años cincuentas y principios de los años sesentas, lo normal era que el hardware fuera de propósito general. Por otra parte, el software se diseñaba a medida para cada aplicación y tenía una distribución relativamente pequeña. El software como producto (programas desarrollados para ser vendidos a uno o más clientes) estaba en su infancia, la mayoría del

---

<sup>14</sup> Pressman Roger. Ingeniería de software. McGraw hill México. 1999. Pag 80



---

software se desarrollaba y era utilizado por la misma persona u organización. La misma persona lo escribía, lo ejecutaba y, si fallaba, lo depuraba. Debido a que la movilidad en el trabajo era baja, los ejecutivos se aseguraban de que esa persona estuviese cuando se encontrara algún error. Debido a este entorno personalizado del software, el diseño era un proceso implícito, realizado en la mente de alguien, y la documentación normalmente no existía.

### **La segunda era del software.**

La segunda era en la evolución de los sistemas de computadora se extiende desde la mitad de la década de los sesenta hasta finales de los setenta. La multiprogramación y los sistemas multiusuario introdujeron nuevos conceptos de interacción *hombre-maquina*. Las técnicas interactivas abrieron un nuevo mundo de aplicaciones y nuevos niveles de sofisticación del hardware y del software. Los sistemas de tiempo real podían recoger, analizar y transformar datos de múltiples fuentes, controlando así los procesos y produciendo salidas en milisegundos en vez de minutos. Los avances en los dispositivos de almacenamiento en línea condujeron a la primera generación de sistemas de gestión de bases de datos.

La segunda era se caracteriza también por el establecimiento del *software como producto* y por el establecimiento de *empresas de software*. El software ya se desarrollaba para tener una amplia distribución en un mercado multidisciplinario. Los programas se distribuían para computadoras grandes y para mini computadoras, a cientos e incluso a miles de usuarios. Los estándares de la industria, del gobierno y de las universidades se disponían a desarrollar el mejor paquete de software y obtener ganancias.

Conforme crecía el número de sistemas informáticos, las casas comerciales desarrollaban proyectos en los que se producían programas de decenas de miles de licencias fuente. Los productos de software comprados al exterior incorporaban cientos de miles de nuevas sentencias. Todos esos programas, todas esas sentencias fuente tenían que ser corregidos cuando se detectaban fallos, modificados cuando cambiaban los requisitos de los usuarios o adaptados a nuevos dispositivos hardware que se adquirían día a día. Estas actividades se llamaron colectivamente *mantenimiento del software*. El esfuerzo gastado en el *mantenimiento del software comenzó a absorber recursos en una medida alarmante*. Aun peor, la naturaleza personalizada de muchos programas los hacía virtualmente imposibles de mantener. Había iniciado una *crisis del software*.



### **La Tercera era del software.**

La tercera era en la evolución de los sistemas de computadora comenzó a mediados de los años setenta y continúa mas haya de una década. El sistema distribuido, múltiples computadoras, cada una ejecutando funciones concurrentemente y comunicándose con alguna otra, incremento notablemente la complejidad de los sistemas informáticos. Las redes de área local y de área global, las comunicaciones digitales de alto ancho de banda y la creciente demanda de acceso instantáneo a los datos, supusieron una fuerte presión sobre los desarrolladores del software. Aún mas, los sistemas y el software que lo permitían continuaron residiendo dentro de la industria y de la academia.

La finalización de la tercera era se caracterizo por la llegada y amplio uso de los microprocesadores. El microprocesador ha producido un extenso grupo de productos inteligentes, desde automóviles hasta hornos microondas, desde robots industriales a equipos de diagnósticos de suero sanguíneo. Pero ninguno ha sido más importante que la computadora personal. En menos de una década, las computadoras llegaron a ser fácilmente accesibles a un mayor número de gentes.

### **La cuarta era del software.**

La cuarta era de la evolución de sistemas informáticos se aleja de las computadoras individuales y de los programas de computadoras, dirigiéndose al *impacto colectivo de las computadoras y del software*. Computadoras personales mas desarrolladas controladas por sistemas operativos sofisticados, en redes globales y locales, acompañadas por aplicaciones de software avanzadas se han convertido en la norma. Las arquitecturas informáticas están cambiando de entornos centralizados de grandes computadoras a entornos descentralizados cliente/servidor. De hecho Internet se puede observar como un software al que pueden acceder usuarios individuales.

La industria del software ya es la cuna de la economía del mundo. Las decisiones tomadas por gigantes de la industria tales como Microsoft arriesgan billones de dólares. A medida que la cuarta generación progresa, comienzan a surgir nuevas tecnologías. Aunque las predicciones de las computadoras de quinta generación continúan eludiéndonos, las técnicas de cuarta generación para el desarrollo del software están cambiando en la forma en que la comunidad del software construye programas informáticos. *Los sistemas expertos y el software de inteligencia artificial han salido del laboratorio para entrar en aplicaciones prácticas de una gran variedad de problemas del mundo real. El software de redes*



neuronales artificiales junto con la aplicación de lógica difusa ha abierto posibilidades excitantes para el reconocimiento de patrones y habilidades de procesamiento de información de carácter humano. La programación de realidad virtual y los sistemas multimedia ofrecen formas radicalmente diferentes de comunicar información al usuario final. Los algoritmos genéticos ofrecen el potencial para el software que reside dentro de las computadoras biológicas masivamente en paralelo.

Sin embargo, un conjunto de problemas relacionados con el software ha persistido a través de la evolución de los sistemas basados en computadora, y estos problemas continúan aumentando. Como es el aumento en el nivel de papeles que se han empezado a atribuir al personal que se encarga de este desarrollo de software, desarrollos cada vez mas complejos, que requieren de grandes recursos para la finalización de proyectos<sup>15</sup>.

El desarrollo de software ha tenido una rápida evolución, en el que se han ido estableciendo etapas que deben cumplirse en cada desarrollo de software, estas han cambiado o se han incrementado, de estas etapas de desarrollo de software se pueden generalizar algunas de ellas, que se exponen a continuación.

#### 1.4.4. DESARROLLO DE SOFTWARE

El desarrollo de software consiste en varias etapas, que dependen de quien lo realice, pues la gran mayoría de quien desarrolla software, propone su forma de trabajar o diseña su propio proceso. Pero de forma general las etapas de un desarrollo de software consisten de las siguientes etapas:

- Análisis y definición de los requerimientos.
- Diseño del sistema.
- Diseño de programas
- Escritura de los programas (Codificación).
- Prueba unitaria y de integración.
- Prueba de sistema.
- Prueba de aceptación.
- Entrega de sistema y,
- Mantenimiento.

<sup>15</sup> Pressman Roger S., "Ingeniería del software "Un enfoque práctico", McGraw Hill, México. 1998. Pag 5



Estas etapas pueden visualizarse en la figura (Figura 1.3), representa el modelo de desarrollo denominado de cascada, que es el más comúnmente utilizado.

### MODELO DE CASCADA<sup>16</sup>

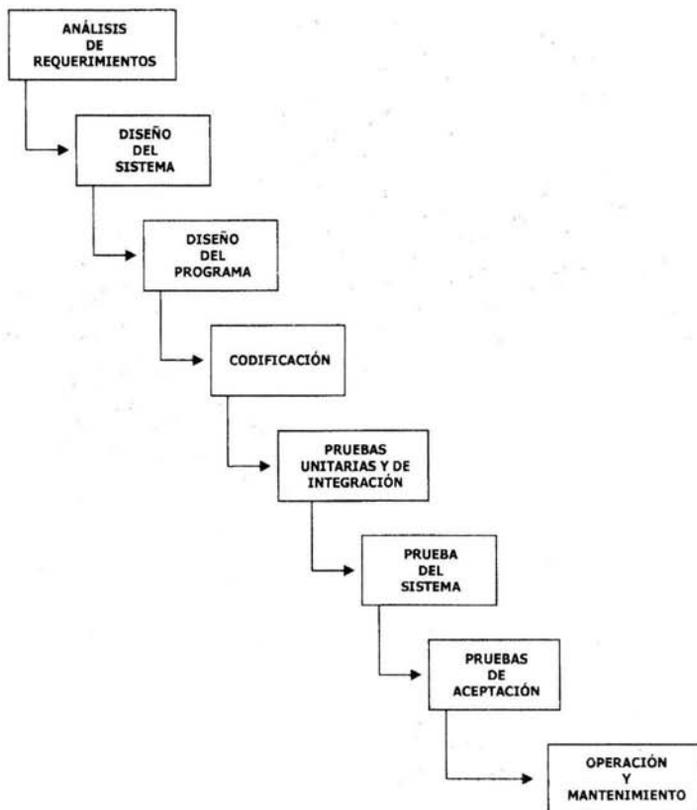


Figura 1.3

Estas etapas de desarrollo, por lo general no se cumplen en el orden listado, ya que cuando no se planean las demás actividades de un proyecto tienen problemas en su finalización y este no es satisfactorio.

<sup>16</sup> Pfeleger Shari Lawrence. "Ingeniería de software" un enfoque practico, Prentice-Hall México, Pag 56



## 1.5 LA IMPORTANCIA DE ADMINISTRAR LOS PROYECTOS DE SOFTWARE

Sin importar la naturaleza específica de un proyecto, el desarrollo de sistemas de software comprende sistemas nuevos o modificados, es decir, implica cambios. La administración eficaz de tales cambios obliga a que los miembros de grupo deban comunicarse de manera satisfactoria, para poder emprender los cambios requeridos a un software o sistema, o el desarrollo de uno nuevo. Esta actividad de desarrollo de sistemas se inicia cuando un individuo o grupo con la capacidad de iniciar cambios en la organización perciben un posible beneficio. Aunque no todas las personas son las responsables de iniciar un proyecto, pues hay limitaciones de jerarquía, poder, autoridad y posición política dentro de la organización. *Lo más frecuente es que los administradores o gerentes sean quienes inicien el cambio, de tal suerte que serán ellos quienes inicien proyectos de desarrollo de sistemas. Además, el grado de apoyo administrativo general, en especial el de alta dirección, influye en forma considerable en las probabilidades de éxito o fracaso de un proyecto de desarrollo de sistemas.*<sup>17</sup>

Las iniciativas planeadas o no, de desarrollo de software surgen de todos niveles de las organizaciones. La planeación adecuada y la participación de los administradores ayuda a garantizar que estas iniciativas den soporte a los objetivos globales de la organización, los proyectos de desarrollo de sistemas generalmente puede ser iniciados por diversas razones:

- Problemas con el sistema existente.
- Interés en aprovechar nuevas oportunidades
- Competencia creciente.
- Interés en hacer un empleo más eficaz en la información.
- Crecimiento organizativo.
- Fusión o adquisición.
- Cambio en el mercado o entorno de los negocios.

Para las organizaciones se pueden tener simultáneamente varios proyectos, los mismos que deben ser administrados, para poder desarrollar un proyecto se deberá elegir el de mayor prioridad, una forma de determinar que proyectos deben ser iniciados es empleando una secuencia de pasos, como el que puede establecerse en el siguiente diagrama (Figura 1.4):

<sup>17</sup> M. Stair, Reynolds, Principios de sistemas de información. Mexico. 2000 Pag. 520-524



### Proceso para la elección de un proyecto de desarrollo de software.<sup>18</sup>

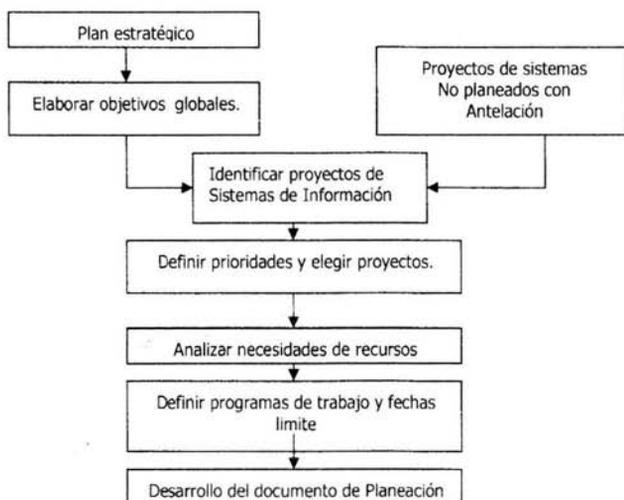


Figura. 1.4

Dependiendo del plan estratégico de la organización, y valuando sus objetivos organizacionales, debe de identificar los posibles proyectos que serán realizados, determinar los recursos con los que se cuenta, determinar el plan de ejecución y desarrollar la documentación requerida para el desarrollo del proyecto.

Hasta este punto del trabajo desarrollado, se ha hecho una serie de definiciones de administración, de software y la forma en que este ha venido evolucionando, se ha expuesto las características del mismo, las complejidades por la que ha pasado y porque se esta en una crisis del software. Esto problemática puede considerarse como una oportunidad, de plantear alternativas de solución, para lograr establecer un diseño de desarrollo que supere las complejidades que se presentan en cada proyecto que es iniciado. Por lo que se realiza un diagnostico de la problemática por la que atraviesa el desarrollo de software.

<sup>18</sup> M. Stair. Reynolds, Principios de sistemas de información. México. 2000. Pag 525



## 1.6 PROBLEMÁTICA EN EL DESARROLLO DE PROYECTOS DE SOFTWARE.

El gran avance tecnológico, ha provocado un importante cambio en las actividades que se realizan en la vida diaria, muchos de estos cambios se han dado en diferentes áreas de conocimiento del hombre, trayendo nuevas formas de comunicación, nuevas formas de hacer negocios, nuevas formas de ganar mercado, hacer transferencias internacionales de información, y un gran número de operaciones que hace pocos años no se hubiese considerado posibles por la mayoría de la población. Este cambio tecnológico ha sido posible a que se han desarrollado proyectos de software, encaminados a hacernos más fácil las actividades diarias.

Este desarrollo de sistemas basados en software, como se presento en la evolución de software, se realizaba con pocas personas cuando apenas iniciaban las primeras computadoras, pero cuando se requirieron adecuaciones y mantenimiento a sistemas ya existentes provoco que se tuvieron que realizar actividades nuevas en cada proyecto.

Al hacerse el desarrollo de software una actividad con un número cada vez mayor de tareas, provoco que estos se hicieran cada vez más robustos, además se tenía un mayor número de requerimientos para mejorar los sistemas existentes de cada software, así como por el tamaño que tiene cada proyecto. Así se inicia una serie de situaciones que no logran controlarse en cada proyecto. Ya que hoy día cada proyecto de software que es iniciado, aun contando con una tecnología adecuada para su realización, *no logran ser finalizados en los tiempos establecidos, elevando con ello sus costos, careciendo de documentación necesaria para poder dar seguimiento a cada proyecto.*

Otra situación que esta relacionada con cada proyecto de software que se emprende, es que los proyectos se realizan de *manera urgente por decisiones de directivos que imponen fechas irreales* de cada proyecto, además el cambio tecnológico requiere que los equipos de computo tengan que actualizarse y la mayoría de las veces no se capacita a su personal, los equipos de computo son obsoletos y los desarrollos ya no son factibles de poderse realizar con esa tecnología.

Algo importante que destacar es que los proyectos que logran realizarse satisfactoriamente deben sus logros en algunos casos a la capacidad del equipo humano que se forma en cada proyecto, ya que el desarrollo de software es una actividad creativa, que se desarrollarse en cada proyecto.



El cambio y la *actualización tecnológica*, ha hecho que las organizaciones tengan que adaptarse a una gran velocidad, que inviertan en gran parte de sus recursos en el desarrollo de nuevos productos, ya que sin actualizaciones tecnológicas, estos cambios no podrán realizarse, provoca que se pueda estar rezagando tecnológicamente, y se pueda estar perdiendo competitividad, provocando que el software que se está desarrollando para un tiempo determinado, en el momento de su implantación el equipo tecnológico ya halla evolucionado, y el software que se ha desarrollado resulte ineficiente, ya que la tecnología la mayoría de las veces no se considera para realización de este tipo de proyectos, así se puede dar el caso de que un proyecto que se inicio en un tiempo determinado, tal vez cuando tenga que implantarse, la tecnología a utilizar ya no este apta para el sistema desarrollado.

Una problemática más que se presenta en los proyectos, es que el personal que es asignado a un proyecto, tiene una *reasignación de proyecto*, provocando con ello un retraso en el proyecto. Al cambiarse o reasignarse personal en un proyecto y *carecerse de documentación* del mismo, provoca además del retraso, que el *costo se eleva*, y su tiempo estimado de finalización se vea alterado.

Resumiendo de los puntos anteriores, se pueden establecer parte de la problemática de los desarrollos de software, se puede considerar que estas causas se deben a una mala administración de proyectos de software. Dentro de las principales causas determinadas anteriormente tenemos lo siguiente:

- No existe planeación para los proyectos.
- El personal es reasignado a proyectos nuevos o a otros ya existentes.
- Falta de capacitación del personal que esta en el proyecto.
- El cambio de tecnología no es previsto durante el proyecto.
- Las decisiones para realizar un proyecto de software, se hacen desde el nivel más alto de la organización, sin tomar participación de quien realiza y conoce la parte de los desarrollos de un software.
- Los costos no son estimados, ni los tiempos de realización, y la mayoría de las veces son rebasados.
- Se carece en la mayoría de los proyectos de documentación suficiente para el proyecto.



Considerando los puntos anteriores, podemos establecer una serie de variables que nos pueden ayudar a proponer alternativas de solución a la problemática planteada. Para ello hay que determinar los factores que provocan esta problemática y plantearlo en términos de variables para poder calificarlos y cuantificarlos, para esto se debe de identificar la naturaleza de cada variable, es decir definir las en términos de si son controlable y no controlable, para que con las que quedamos llegar a controlar, establezcamos las soluciones a la problemática planteado. Ya que una solución podrá darse en términos del planteamiento que podamos hacer respecto de las variables que consideremos, con el cual se pueden proponer alternativas de solución con respecto al manejo de estas. Las variables son consideradas controlables cuando quien toma las decisiones puede controlarlas, las variables concernientes incontrolables pero que pudieran ser controladas si así fuese necesario, a menudo hacen posible la obtención de soluciones creativas y efectivas.<sup>19</sup> Así entonces debe poderse establecer las variables que conciernen a la problemática, para encontrar alternativas de solución.

#### 1.6.1. DEFINICIÓN DE VARIABLES DE LA PROBLEMÁTICA

En la problemática planteada, se han listado a varias causas que están presentes en los proyectos de desarrollo de software, mismas que son planteadas para determinar el nivel de prioridad para cada una de ellas. De estas variables que se manejan en la posible solución de la problemática podemos identificar las siguientes variables, una de ellas es la gestión administrativa, tiempo de finalización, capacitación, equipos de trabajo, documentación, como variables que pueden ser controlables. (Cuadro 1.1)

**Variables de ambiente controlables.**

Jerarquización	Variables Controlables
1	Gestión Administrativa
2	Capacitación
3	Equipo de trabajo
4	Tiempo de finalización

Cuadro 1.1

<sup>19</sup> Ackoff Russel, "El arte de resolver problemas", Limusa, México, 2003. Pag 23



**Gestión Administrativa.** Definida como el proceso de administración, para el desarrollo de proyectos de software, en el que se definen las funciones que a realizar en cada proyecto que tiene que administrarse, contemplan las funciones administrativas propias para desarrollo de software. Como es la definición de proyecto, la planeación, la organización, el control y dirección del proyecto.

**Capacitación.** Se refiere a la capacitación que debe de tener el personal que este involucrado en un proyecto, capacitación que debe contemplar las herramientas tecnológicas de desarrollo de software. Capacitación que se da en términos de las necesidades de cada proyecto que se este desarrollando y con las actitudes que cada persona tenga en el proyecto.

**Equipos de trabajo.** Formación de equipos como forma de organización para cada proyecto de desarrollo, el establecimiento niveles de comunicación entre estos equipos para la elaboración de cada proyecto.

**Tiempo de finalización.** Definido como la estimación que debe realizarse para la elaboración de un plan de desarrollo de software, en el que se consideren todas las actividades, que involucra el ciclo de desarrollo de software.

Hasta el momento se han determinado los variables que pueden ser controladas por quien toma las decisiones en cada proyecto o proyectos a realizar. Debe determinarse también las variables no controlables pero que forman nuestro medio y tiene implicaciones directas durante cada desarrollo de software, una de ellas es la tecnología, la otra es la rotación de personal, los costos, normatividad, estas variables que habrá de considerar para la tomar las decisiones que ayuden proponer soluciones a la problemática previamente planteada. (Cuadro 1.2)

**Variables de ambiente no controlables.**

Número de Jerarquización	Variables
1	Normatividad
2	Tecnología
3	Documentación
4	Rotación de Personal
5	Costos

Cuadro 1.2.



---

**Normatividad.** Son los estándares que establecen organizaciones para normar los procesos a seguir para desarrollar un producto de software, dichas normas son establecidas por estas mismas organizaciones para regular la calidad del producto.

**Tecnología.** Se refiere a los elementos que son necesarios para el proceso de desarrollo de un producto de software, como son las computadoras, software de desarrollo que sirve como herramienta para realizar proyectos de software, es decir para crear software para la propia necesidad de la organización.

**Documentación.** La documentación definida como la forma de comunicación que se puede establecer en el desarrollo de un proyecto. La cual permita la retroalimentación de las actividades y del plan de desarrollo de cada proyecto, sirve como el medio de comunicación más tangible de dar a conocer las actividades que involucra cada proyecto.

**Rotación de personal.** Definida como el personal que es removido, dado de baja de proyecto o cambiado a uno nuevo sin antes haber concluido el previo. Con lo que no termina su función para el proyecto en el que había dado inicio.

**Costos.** Se refiere al costo de la adquisición de equipos de cómputo, al costo de contratación del personal, herramientas de desarrollo del software, como son licencias de software, el costo de la misma planeación que se realiza, costos administrativos y de servicios que se realizan para cada proyecto.

Estas variables con respecto al tiempo, tendrán un comportamiento que puede considerarse para dar alternativas de solución, dependiendo el manejo que se haga de ellas se podrán considerar como variables que nos permitan encontrar alternativas de solución a la problemática. En el siguiente cuadro plantea las las posibles tendencias que pueden tener con respecto al tiempo (Cuadro 1.3)



### Comportamiento de las Variables respecto al tiempo

Variables	A corto plazo	A mediano plazo	A largo Plazo
Gestión administrativa	Controlable	Controlable	Controlable
Normatividad	No controlable	No controlable	No controlable
Tecnología	Controlable	No controlable	No controlable
Capacitación	Controlable	Controlable	Controlable
Documentación.	No controlable	Controlable	Controlable
Tiempo	Controlable	No Controlable	Controlable
Equipos trabajo	Controlable	Controlable	Controlable
Rotación de personal	No Controlable	No controlable	No controlable
Costos	No Controlable	Controlable	Controlable

Cuadro 1.3

#### PROCESO ADMINISTRATIVO.

Administrativamente los proyectos en las organizaciones donde se desarrolla software, no siempre una organización bien definida, se carece de planeación para cada proyecto realizado, se establecen los papeles de trabajo para personas que no tienen las habilidades para el proyecto que se esta creando, no existen metodologías de desarrollo para crear la codificación de un sistema de software, no hay registro de los proyectos realizados con previamente, al no existir un registro de los proyectos iniciados y finalizados, no se tiene retroalimentación de lo que se ha realizado. No se registra la tecnología con la que se cuenta, se carece de herramientas de desarrollo que den seguimiento, a cada nuevo proyecto de software. La administración no es eficiente, y por lo mismo la función de la planeación no aparece.



## **1.7 CONCLUSIONES.**

El desarrollo de software es una actividad de apenas cinco décadas, que ha tenido un rápido desarrollo con los grandes avances y cambios tecnológicos, estos cambios requieren de personas cada vez más especializadas, para cada nueva actividad dentro de los desarrollos software. El manejo de estas actividades solo se lograra realizar a través de una administración eficiente, ya que como se ha expuesto en la sección 1.1 referente a la administración, esta debe cumplir las funciones administrativas necesarias en cada proyecto realizado. Ya que hoy los proyectos que se realizan tienen debilidades en mayor porcentaje del tipo administrativo y en menor medida de tipo técnico, por lo tanto se puede decir que todo proyecto requiere en mayor medida de procesos de tipo administrativo para poder lograr un proyecto de forma exitosa.

El entender la administración, nos facilita su realización y se puede manejar de una manera más adecuada, ayuda a quien la realiza, a tomar las decisiones más acertadas, ya que la administración como se definió en el sección 1.2 tiene funciones que deben ser comprendidas para su aplicación. Por lo que entender las funciones administrativas no crea expertos administradores, pero da herramientas para poder crear nuestra propia forma de administrar proyectos y poder tomar las decisiones más concretas en cada proyecto, ya que la administración es tanto una ciencia como un arte que habrá de realizar.

El desarrollo de software, requiere de procesos de administración, que logren afrontar la problemática que existe hoy día, ya que como se planteo en las etapas de evolución de software no es una problemática actual, sino un problema que se empezó a gestar, cuando se realizaron proyectos con un mayor alcance comercial, por ello debe buscarse modelos de administración que asegure la creación de software en tiempos determinados, con los recursos necesarios y suficientes, tal y como se pudo determinar en la problemática afrontada en la sección 1.4 referente a la evolución de software.

Una estructura organizacional bien definida puede ayudar al logro de los proyectos, pero una mala organización puede provocar la no finalización de un proyecto, la elección de una estructura organizacional es importante, ya que de ella se derivan varias acciones a seguir en cada proyecto, la ventaja de definir una estructura organizacional al iniciar el proyecto, es el tener un mayor control sobre las tareas y actividades a realizar, así cada



---

etapa del proyecto se realizada con una administración de forma eficiente.

Administrar proyectos de software es una actividad que necesariamente se realiza aun y sin conocimiento de que es la administración, quien únicamente se ha dedicado a la parte del software a nivel técnico, la administración de proyectos es la actividad que en algún momento habrá de realizar, es decir pasara del nivel técnico a un nivel administrativo, por eso es importante, comprenderla para desarrolla una administración eficaz y eficiente, visualizando las complejidades por las que pasa el software y poder determinar el curso de acción para cada proyecto. Tal y como se plantea en la punto referente a sección 1.5 de la importancia de administrar proyectos de software, que hoy es una de las actividades mas demandantes.

Al estar fuertemente relacionado el software con la tecnología y formar parte misma de ella, se impone una velocidad de cambio, obligando a tomar decisiones en un tiempo muy corto, pero la falta de planeación de un proyecto de software, una mala preparación del personal, la falta de capacitación, la mala estimación de los costos, seguirá siendo una constante problemática que como se presento en la sección 1.6 del capitulo, esta problemática debe ser afrontada, para encontrar soluciones que ayuden a desarrollar proyectos que superen su problemática y lograr su finalización con los recursos, en los tiempos y presupuestos previamente establecidos.



---

## FUENTES DE CONSULTA

Ackoff Russell L.,

El arte de resolver problemas.

Limusa. México. 2003

Kendall y Kendall.,

"Análisis y Diseño de Sistemas",

Prentice-Hall, México, 1994

Pfleeger Shari Lawrence

Ingeniería de software

Teoría y Práctica

Prentice Hall 2002

Pressman Roger S.

Ingeniería del software "Un enfoque práctico"

McGraw Hill, México. 1998

Ralph M.Stair. Reynolds George W.,

Principios de sistemas de información

Thomson editores. México. 2000

Reyes Ponce Agustín,

Administración Moderna.

Limusa. Mexico 2001

[www.monografias.com](http://www.monografias.com) "Historia de la computación".

[www.alpha-hardin.com](http://www.alpha-hardin.com)

Roberto Toledo. Administre mejor sus proyectos, 12 pasos básicos



---

## CAPITULO 2.

### LA PLANEACION DEL DISEÑO DE PROYECTOS DE SOFTWARE

Objetivo: Elaborar un diseño de proyecto para el desarrollo de software a la medida, que permita establecer las principales líneas de acción, e involucre los elementos necesarios para dar seguimiento a los proyectos de software, que asegure la finalización de los proyectos en los tiempos y costos estimados.

En el capítulo precedente se ha realizado un diagnóstico de la problemática que presenta los proyectos de software, se ha considerado las principales causas que provocan la problemática, estas causas son las que se tienen que afrontar para proponer soluciones.

La propuesta de solución que se presenta como resultado del diagnóstico hecho previamente planteada, esta basada en el desarrollo de un diseño administrativo que permita manejar eficientemente cada proyecto, que logre como objetivo principal finalizar en los tiempos y dentro del presupuesto establecidos. Para ello se propone la utilización de la administración de proyectos, ya que la administración de un proyecto consiste, en terminar cada proyecto iniciado, cumpliendo los requerimientos, con las personas y los presupuestos previamente planteados.

Para el desarrollo del diseño que administre proyectos, debe hacerse con un propósito prospectivo, en el que estemos hablando del futuro en el presente como establece Miklos, en el que los futuros deben ser probables y posibles, para cada situación que se presente, se debe de aceptar la incertidumbre como parte esencial del diseño y como un elemento más de trabajo.<sup>20</sup> El futuro es el resultado de acciones sobre las que en ocasiones puede tener influencia directa o tangencial significativa. A si el diseño que se realiza, a decir de Ackoff es en parte previsible, diseñable y construible.<sup>21</sup>

Para la elaboración del diseño que de elementos de solución a la problemática planteada, se deben definir los fines y los medios que serán utilizados para la elaboración del diseño desarrollo software.

---

<sup>20</sup> Miklos Tomas, Las decisiones políticas, de la planeación a la acción. Siglo XXI, México 2003. Pag 31.

<sup>21</sup> *Ibid* 32.



Las organizaciones que realizan proyectos de software a la medida, es decir para su propio uso, definen sus procesos de administración para sus proyectos, pero cuando este proceso no se realiza, se dice que la organización tiene una debilidad organizacional, provocando un manejo deficiente de sus recursos, por ello es importante plantear el tipo de organización a seguir en la administración de proyectos.

Para hacer que los proyectos se terminen dentro de un presupuesto definido, en un tiempo estimado, con los requerimientos previamente establecidos, debe considerarse las actividades que se realizarán en el proyecto, así cuando se da inicio a un proyecto se determinan las consideraciones que tendrá el proyecto, debe cumplir las condiciones que tienen el desarrollo de software, dichas condiciones estarán determinadas por lo siguiente:

- Se establece una fase de inicio para el proyecto.
- Se determinan tiempos de inicio y finalización de cada proyecto, apoyados de herramientas tecnológicas, que faciliten la realización de esta tarea.
- Se definirá la organización, para la realización de actividades, que faciliten su desempeño, y que se logre una comunicación entre sus participantes.
- Debe elegirse las herramientas de desarrollo existentes para la creación de software, herramientas como lenguajes de programación.
- Considerar las habilidades del personal, para la formación de equipos de trabajo.
- Elegir un plan de documentación que registre las actividades del proyecto.
- Elegir la forma en que se dará seguimiento al proyecto.
- Elegir alguna metodología para el ciclo de desarrollo de software.

Al definir los pasos a seguir para el diseño ideal, estamos definiendo los fines que se cubrirá el diseño ideal, al ser el diseño un sistema de administración, éste considerará las tres responsabilidades interdependientes establecidas por Ackoff, tales responsabilidades determinan lo siguiente: La primera responsabilidad es hacia los propósitos del sistema que administra (control); la segunda hacia los propósitos de las personas que forman parte del sistema administrado (humanización); y la tercera hacia los propósitos del sistema que los abarca y de los demás sistemas que lo contenga (ambientalización).<sup>22</sup>

---

<sup>22</sup> Ackoff Russel, "El paradigma de Ackoff", Una administración sistémica". Ed Limusa, México, 2002. Pag 273-275



La administración consistirá de varias etapas, que se consideran para la elaboración del proyecto, determinará las actividades a seguir, para que el proyecto se realice y se lleve a cabo con una administración eficiente, que logre cumplir los fines para los que se está diseñando, en el que el factor humano debe considerarse como el elemento clave para la realización de los mismos.

Las etapas que deben seguirse en un proyecto de software son las que a continuación se describen, siendo éstas una de las principales para la administración de los proyectos:

## **2.1 PREPARACION DEL PROYECTO**

Las actividades pueden variar dependiendo de la organización y del tamaño del software a desarrollar, sin embargo las actividades que se realizan para la preparación de un proyecto son las siguientes:

- Redacción de la propuesta.
- Planeación y calendarización del proyecto.
- Costeo del proyecto.
- Selección y evaluación de personal.
- Supervisión y revisión del proyecto.
- Redacción y presentación de informes.

Para iniciar el proyecto de software, el primer punto es la redacción de la propuesta, implica describir los objetivos que persigue el proyecto, descripción de los principales detalles que se espera de este, la forma en que se pretende realizar, incluye estimados de costo y un programa de actividades que pueden ser realizadas. Justifica el por qué se requiere el desarrollo, estableciendo si el contrato se otorgara a una organización en particular o a un equipo de trabajo. La redacción de la propuesta puede ser una tarea crítica, ya que la redacción de propuestas es una habilidad que se adquiere con la experiencia, esta propuesta es la que determina si se realiza o no el proyecto.

El siguiente punto que se refiere a la planeación del proyecto, identifica las actividades y entregas producidas del proyecto, por lo tanto se debe de bosquejar un plan para guiar el desarrollo hacia las metas establecidas.



La estimación de costo es una actividad relacionada a la definición del proyecto, y una de las más importantes para poder iniciar un proyecto, esta estimación contempla el estimado de los recursos requeridos para llevar a cabo el plan del proyecto, tales recursos pueden ser los equipos de trabajo, los edificios, las herramientas para la elaboración de programas, el personal que se requerirá, el hardware de desarrollo, además de los recursos necesarios para el tipo de proyecto que se realice.

La supervisión del proyecto, se realiza por el administrador, siendo esta una actividad continua durante el desarrollo del proyecto. El administrador debe de tener conocimientos del progreso del proyecto, comparar los progresos y costos reales con los planeados. Para esto debe tener mecanismos formales para supervisar los avances, un administrador hábil logra una imagen clara del proyecto, con una entrevista informal con el personal que este en el proyecto, con ello puede determinar el avance del mismo. Durante un proyecto es normal hacer varias revisiones formales de su administración, para determinar el progreso de los desarrollos técnicos del proyecto, tomando en cuenta el estado que guarda el proyecto, determinar si se están cumpliendo los propósitos de la organización.

El tiempo de desarrollo para un proyecto de software puede ser de varios años, durante ese tiempo, los objetivos de la organización tienden a cambiar, esos cambios pueden significar que el software ya no se requiere más o que los requerimientos originales del proyecto son inapropiados. La administración puede decidir parar el desarrollo del software o cambiar el proyecto para adecuarlo a los cambio de los objetivos de la organización. Por ello, debe establecerse en el plan de proyecto, el estimado de tiempo de cada actividad que se esté considerando.

Parte importante es el recurso humano que forma parte de los proyectos, por ello la selección que se haga del personal, debe de basarse en la habilidad y experiencia que tenga el personal para el proyecto. Sin embargo, es responsabilidad del administrador, formar el equipo ideal mínimo para cada proyecto, considerando las habilidades y capacidades de cada elemento humano que conforme un equipo de trabajo de acuerdo a los requerimientos que se presenten para cada desarrollo de software.



Un administrador del proyecto, es el responsable de dar informes del mismo tanto al cliente como a las organizaciones, dichos administradores deben redactar documentos concisos y coherentes que resuman la información crítica de los informes detallados del proyecto. Les debe ser posible presentar esta información durante las revisiones de progreso, en consecuencia, comunicarse efectivamente de forma oral y escrita es un habilidad que un administrador de proyectos debe de tener, para poder controlar el mismo.

Hasta el momento se ha definido las etapas para preparar un proyecto, independientemente del tamaño, estas etapas son las mínimas necesarias para cada proyecto que es elaborado. Cada etapa debe especificarse en tiempos y actividades tendientes a ser realizadas independientemente del tamaño organizacional que pueda tener el proyecto, la definición establece y determina el inicio del mismo.

## **2.2 DEFINICION DEL PROYECTO**

Al iniciar el proyecto se define el objetivo que se persigue con la elaboración del proyecto, pues este es el punto de partida del desarrollo, de la problemática planteada, la definición de este objetivo se logra en base a reuniones con los usuarios que serán quienes utilicen el software desarrollado. En una reunión puede determinarse el objetivo del proyecto, los recursos para el desarrollo, las consideraciones y restricciones que tenga el proyecto. Se hace participe al personal que intervendrá directamente, además se determina la organización que se usara. Así para que se de inicio, deben de quedar definidos los siguientes puntos:

- El enunciado del problema.
- La organización que debe tomar.
- Las actividades preeliminarias que se contemplaran.

Habiendo formulado el enunciado del problema, se registraran las tareas ha desarrollar, determinando el orden cronológico de ejecución, los tiempos estimados que llevara la realización cada actividad, estos tiempos generalmente son estimados por alguna persona que ha realizado proyectos anteriores o ha estado involucrado en alguno, si se cuenta con algún registro de otros proyectos estos ayudaran a tener una estimación mas adecuada para el que dará inicio.



La estimación de los tiempos sigue siendo una de las principales problemáticas de cada proyecto, ya que cada proyecto es único y este establece su propia forma de estimarlos, algunas de las veces el tiempo de realización de un proyecto puede depender de la habilidad que pueda tener un equipo de trabajo durante el desarrollo.

Como etapa posterior a elaboración del enunciado, debe diseñarse la organización a seguir, ya que la organización puede determinar en gran medida el buen desarrollo del proceso.

### **2.3 ORGANIZACIÓN.**

La organización a considerar en un proyecto, puede determinar la finalización oportuna, para ello debe de elegirse la que permita un nivel de comunicación flexible con sus integrantes, considerando además el tamaño que debe tener, ya que hay proyectos que se pueden realizar con únicamente tres personas, y habrá otros que tengan que realizarse con varias decenas. Las personas que sean consideradas en la organización, deberán estar relacionados y coordinados. La elección de la mejor organización dependerá de los siguientes aspectos:

- Los antecedentes y estilos de trabajo de los miembros del equipo.
- El número de personas con el que se forme el equipo de trabajo.
- Los estilos de administración de los clientes (usuarios) y desarrolladores.

#### **2.3.1 ATRIBUTOS DE LAS PERSONAS PARA FORMAR EQUIPOS.**

La formación de equipos es una forma de organización para emprender un proyecto, estos equipos están formados por personas relacionadas de una u otra manera con el proyecto que se desarrollara, ya sea por experiencia o porque tiene los conocimientos que requiere para poder formar parte del equipo de trabajo. Así para formar un equipo de trabajo es importante entender los atributos que tienen las personas para desarrollar sus actividades.



Las personas tenemos diferentes formas de interactuar y de realizar nuestras decisiones, ya que para algunas personas es preferible hacer un análisis antes de tomar una decisión, y otras confían en su intuición para tomar la mayoría de las decisiones importantes. Se puede pensar en el estilo de trabajo preferido en términos de dos componentes, la forma en la cual se comunican los pensamientos y se reúnen las ideas, y el grado en que las propias emociones afectan a las decisiones a tomar. Cuando se comunican ideas, algunas personas les transmiten a otros pensamientos, y otras piden sugerencias de otras personas antes de formar una propia opinión. A los primeros se les denomina extrovertidos e introvertidos a los últimos. Esto afecta la forma de interactuar entre los proyectos que se pudieran estar desarrollando. Asumimos que se puede catalogar al personal como intuitivo, cuando basa sus decisiones en sentimientos y reacciones emocionales hacia un problema. Otras personas son racionales, deciden principalmente observando los hechos y considerando cuidadosamente todas las opciones.

Este tipo de características en las personas nos permite, describir una variedad de opciones estilos de trabajo, donde los estilos de comunicación constituyen el eje horizontal, y los estilos de decisión el vertical. Mientras más extrovertido se es, el estilo de trabajo cae en la parte derecha del gráfico. Asimismo mientras más emociones toman la parte de las decisiones, más arriba está. Por lo tanto se pueden definir cuatro estilos de trabajo, esto se puede apreciar en la figura (Figura 2.1). Las características de la persona entonces pueden definirse según los atributos con los que cuenta cada persona.

#### **Los extrovertidos racionales.**

- Tienen a mantener sus ideas y no permitir que una intuición afecte las decisiones a tomar.
- Les dicen a sus colegas lo que los colegas quieren saber, pero raramente piden más información.
- Cuando razonan, confían en la lógica no en la emoción.

#### **Los introvertidos racionales.**

- Evitan decisiones emocionales.
- Dispuestos a tomarse tiempo para considerar todos los posibles cursos de acción.
- Son acopiadores de información: no se sienten seguros de tomar una decisión a menos que se estén convencidos de tener todos los hechos a la mano.



### Los extrovertidos intuitivos

- Basan muchas decisiones en reacciones emocionales y tienden a querer comentar con los demás sobre estas, más que a pedir opiniones.
- Usan su intuición para ser creativos y a menudo sugieren aproximaciones no usuales para solucionar problema.

### El introvertido intuitivo.

- Es creativo, pero aplica la creatividad solo después de haber reunido suficiente información sobre la cual basa su decisión.

### Estilos de trabajo de personal<sup>23</sup>



Figura 2.1

Los atributos en cada persona, nos permite el formar equipos de desarrollo que logren una participación activa en cada proyecto, nos permite elegir el mejor equipo de desarrollo, pues al saber el tipo de persona con el que se cuenta, se puede afrontar diferentes situaciones que tenga cada proyecto.

<sup>23</sup> Pileeger Lawrence Shari, "Ingeniería de software", Teoría y Práctica, Prentice Hall 2002. Pag. 107-110



### 2.3.2 EL NÚMERO DE INTEGRANTES PARA FORMAR UN EQUIPO.

Las organizaciones dedicadas al desarrollo de software forman equipos de trabajo, quienes aseguran la calidad del producto, y tomando en cuenta que las reuniones de equipos son una herramienta significativa para la recopilación del estado en el que se encuentra el proyecto, la negociación y la toma de decisiones, las restricciones en cantidad de participantes en las reuniones son el límite superior para el tamaño del equipos se plantea de la siguiente manera: <sup>24</sup>

**Tres miembros.** Implica que durante las reuniones cada miembro tiene oportunidad de hablar. Los asuntos se discuten a profundidad y se resuelven con facilidad. Un posible problema con este tamaño el equipo es que un miembro puede dominar a los otros dos. Además, los equipos pequeños sufren la acumulación de varios papeles para cada participante.

**Cuatro miembros.** Al iniciar con equipos de este tamaño se permite que siga funcionando aunque algún participante salga del equipo. Sin embargo el término de un consenso con un equipo de miembros pares puede ser problemática, ya que puede haber empate en las votaciones. En consecuencia la resolución de problemas puede llevarse relativamente mucho más tiempo.

**Cinco a seis miembros.** Este es el tamaño ideal para un proyecto de desarrollo de software. Los miembros todavía pueden reunirse frente a frente. Las diversas perspectivas, soportadas por una mezcla de ideas, opiniones y actitudes, promueven el pensamiento creativo. Se pueden asignar papeles únicos a cada miembro, creando una mini organización en la que cada miembro complementa a los demás.

**Siete miembros.** Este es todavía un tamaño de equipo relativamente efectivo, pero las reuniones de equipo tienden a ser largas. Cuando se integran equipos de siete integrantes se deben crear subequipos que ayuden a solucionar los asuntos pendientes de la agenda.

---

<sup>24</sup> Bruegge, Dutoit. Ingeniería de software orientada a objetos. Prentice-Hall. Mexico 2002. Pag 451-453



**Ocho o más miembros.** Los equipos de este tamaño llegan a ser difíciles de manejar. La estructura interna con frecuencia comienza a dividirse en subequipos. A menudo ocurren coaliciones y conversaciones laterales durante las reuniones formales. Los miembros tienen más oportunidades de competir o cooperar. Aunque los resultados son satisfactorios, por lo general el tiempo para alcanzar resultados satisfactorios es más largo que en un equipo de menor tamaño.

Las personas en el equipo de trabajo desarrollan actividades una vez formado el equipo, así también adquieren responsabilidades para con el proyecto, estas son establecidas en función del proyecto que se está realizando, así las responsabilidades mínimas serán de siguiente tipo.

### 2.3.3 LOS PERSONAS INVOLUCRADAS EN EL PROYECTO.

El proceso de software lo componen participantes que pueden clasificarse en una de cinco categorías, las cuales son las principales responsabilidades y papeles de cada persona.<sup>25</sup>

- a. Gestores Superiores: que definen los aspectos de negocios que a menudo tienen una significativa influencia en el proyecto.
- b. Gestores (técnicos) del proyecto: que deben planificar, motivar, organizar y controlar a los profesionales que realizan el trabajo de software.
- c. Profesionales: que proporcionan las capacidades técnicas necesarias para la ingeniería de un producto o aplicación.
- d. Clientes, que especifican los requisitos para la ingeniería del software.
- e. Usuarios finales, que interactúan con el software una vez que se ha entregado para la producción.

Estas responsabilidades no son todas, ni son las únicas, pero son las que debe por lo menos ser consideradas en cada proyecto, y que estarán presente de alguna forma en el mismo. Para ser eficaz, el equipo del proyecto debe organizarse de manera que considere las habilidades y capacidades de cada persona. Esta es una tarea que recae principalmente en los jefes de equipo.

---

<sup>25</sup> Pressman Roger S. Ingeniería de software. Un enfoque práctico. McGraw-hill. México. 1999. Pág. 41



---

### 2.3.4 ASIGNACION DE RECURSO HUMANO AL PROYECTO

Los atributos de las personas nos permiten identificar estilos de trabajo, lo que ayuda a conformar equipos que responden a las necesidades para las cuales se les asigna una actividad, entre otras características a considerar es la comunicación para desarrollar un desempeño óptimo en el equipo, al realizar la asignación de actividades se debe considerar lo siguiente<sup>26</sup>:

- Capacidad para desempeñar el trabajo.
- Interés en el trabajo.
- Experiencia con sistemas similares.
- Experiencia con herramientas o lenguajes similares.
- Experiencia con técnicas de desarrollo de sistemas de software.
- Experiencia con ambientes de desarrollo similar.
- Habilidad de comunicación.
- Habilidad para compartir responsabilidad con otros.
- Capacidad de gerenciamiento.

El personal que integra el proyecto será dirigido, supervisado por personas que tengan la capacidad de dirección y supervisión, quienes determinan la estructura organizacional que debe utilizarse, para lograr la mayor certeza, estabilidad, uniformidad en el producto final, que será el de mayor calidad, el más adecuada dependiendo tamaño del proyecto que se desarrolle. La estructura organizacional jerárquica puede ayudar a la organización, debido a se requiere de menos comunicación entre sus integrantes del proyecto cuando este es de pequeño alcance, de este modo las dudas que pudieran surgir durante el desarrollo se revisan entre el jefe de programadores y el programador que es quien desarrolla la codificación del sistema, dependiendo del tamaño del proyecto que este en desarrollo.



## **2.4 COMUNICACIÓN EN EL PROYECTOS.**

En cada proyecto los integrantes del mismo, se comunican unos con otros, con los usuarios y con los clientes. Si la comunicación no es la adecuada puede causar fallas para el proyecto y afectar la calidad del producto resultante. Para ello se considera que entre más personas integren el equipo de trabajo, más líneas de comunicación se pueden tener entre los individuos del proyecto. De forma general si un proyecto tiene  $n$  personas, hay  $n(n - 1) / 2$  pares de personas que podrían poder comunicarse y  $2n - 1$  grupos posibles que pueden crearse para trabajar en secciones más pequeñas del proyecto. Esto puede ser una comunicación de forma verbal, pero existen otro tipo de comunicación como es la escrita, que es una de las partes mas importantes del proyecto, pues es la que genera la documentación, en el cual se describe lo que se planea hacer y lo que se realiza en cada proyecto.

La comunicación es importante, pues permite la retroalimentación entre el equipo de trabajo y los participantes del proyecto, puede realizarse por medio de comunicados, que enteren al personal el estatus que guarda el proyecto, y hacer partícipes a todos los integrantes del proyecto, una forma de hacerlo, es mantener una comunicación entre las personas de un proyecto son las reuniones, pues en ellas se pueden expresar puntos de vista, realizar consensos y tomar las decisiones pertinentes para proyecto.

## **2.5 LOS RECURSOS MATERIALES Y COSTOS DEL PROYECTO.**

Los recursos en los proyectos de software son esenciales y puede determinar el éxito del mismo. Entre los recursos que se consideran para los proyectos de software, es el Hardware, el Software de desarrollo, el recurso humanos y los recursos financieros, ya que sin estos un proyecto de software no puede realizarse.

El hardware es importante para el desarrollo de software, ya que es la herramienta indispensable para el desarrollo de un nuevo software o el mantenimiento de uno existente, ya que sin esta herramienta el desarrollo del software no puede ser sometido a pruebas.

---

<sup>26</sup> Braude Ingeniería de software. Teoría y práctica. Prentice Hall. Brasil. 2002. Pag. 104



El presupuesto, es otro de los recursos que puede llegar a determinar si se inicia un proyecto, ya que se puede tener un plan establecido, pero si no se tiene el recurso económico tal vez no pueda dar inicio. El presupuesto incluye distintos tipos de costos que deben ser considerados, como son: costos de instalación, personal, métodos, herramientas. Los costos de instalación incluyen hardware, espacio físico, mobiliario, teléfonos, módems, calefacción y aire acondicionado, cables, discos, papel, lápices, fotocopiadoras y todos los otros elementos que forman parte ambiente físico, en el cual los desarrolladores trabajaran. Los costos pueden ser estimados, y pueden llega a variar respecto a la estimación inicial. Ya que para proyectos nuevos habrá que considerar la ubicación de oficinas, las condiciones del piso para colocar equipo de cómputo, los controles de temperatura que deben tener, todo lo que haga posible el desarrollo del proyecto.

De los costos que son propios del desarrollo de software, se tiene que considerar los costos que involucran adquisición de software para desarrollo, la adquisición de licencias de software. Además de herramientas para el diseño y la codificación del sistema, el proyecto puede adquirir software para capturar los requerimientos, organizar la documentación, probar la codificación, dar seguimientos a los cambios, generar datos de prueba, dar soporte a las reuniones de grupos, etc.

### **2.5.1 COSTOS DEL PROYECTO.**

En el inicio de los desarrollos de software, los costos resultaban no ser de gran impacto como en la actualidad, ya que lo que resultaba con un costo muy elevado cuando se dio inicio de los proyectos fue el hardware, esto es el equipo necesario para realizar la programación del software, pero hoy día en un proyecto el costo de desarrollo de software representa casi el 80% de los costos totales del proyecto. Por lo que las estimaciones pueden marcar la diferencia entre beneficios y pérdidas. Sobrepassarse en el coste puede ser de gran impacto para el equipo de desarrollo, ya que muchas veces implica el que se realice o no el proyecto.

Los costos, incluyen el costo para desarrollar el software, entregarlo y administrarlo, además los costos no solo incluye consideraciones de diseño sino también las administrativas. Cuando el sistema reemplaza a otro antiguo, se toma en cuenta el costo de asegurar la compatibilidad retrospectiva o la transición hacia el nuevo sistema. Deben de contemplarse costos de entrenamiento a los usuarios finales, los costos de transición y los costos de mantenimiento. El mantenimiento de la compatibilidad retrospectiva con un



sistema anterior puede sumarse al costo de desarrollo, reduciendo el costo de transición. El cuadro siguiente muestra los costos que involucran el desarrollo y mantenimiento de software. (Cuadro 2.3)

**Costos que considera el desarrollo de software<sup>27</sup>**

<b>Criterio</b>	<b>Definición</b>
Costo de desarrollo	Costo del desarrollo del sistema inicial
Costo de entrega	Costo de la instalación del sistema y del entrenamiento a los usuarios
Costo de actualización	Costo de trasladar los datos del sistema anterior. Este criterio es resultado de los requerimientos de compatibilidad retrospectiva
Costo de mantenimiento	Costo requerido para la corrección de errores y para las mejoras al sistema
Costo de administración	Dinero requerido para la administración del sistema

Cuadro 2.3

## **2.6 DOCUMENTACIÓN DEL PROYECTO**

La documentación es una de las principales formas de poder retroalimentar a las personas que están en un proyecto. Muchas de los desarrollos de software, no consideran en la mayoría de los proyectos la importancia de ser documentados, pero la documentación es una de las formas de poder comunicar y entender cada proyecto en su esencia, la metodología que se emplea para cada proyecto, la forma de hacer revisiones, o dar mantenimiento a un proyecto ya finalizado. Los proyectos bien documentados permiten hacer una retroalimentación y servir de apoyo a otros proyectos, y para dar seguimiento al que se esta desarrollando, ya que con su documentación puede ser retomado por personas que se integran al proyecto ya iniciado, hacen más rápida la integración del personal, permiten saber los objetivos que se persiguen.

<sup>27</sup> Bruegge, Dutoit, Ingeniería de software orientada a objetos, Prentice-Hall México 2002 pag. 194



Así la documentación es una forma de transmitir información hacia el personal que desarrolla y administra proyectos de software. Evita la pérdida de tiempo innecesario en el entendimiento de los proyectos ya desarrollados, o en proceso de desarrollo, y sirve como antecedente para proyectos futuros. El tener una documentación para dar seguimiento a los procesos establecidos, para no depender de las personas que desarrollan los sistemas que no documentan sus desarrollos.

Por la importancia que tiene la documentación de proyectos de desarrollo de software, instituciones como la IEEE (Instituto de Ingenieros Electronicos y Electricos) han establecido una norma que dicta los requerimientos que debe considerar un proyecto de desarrollo de software. Dicha norma establece varias secciones que serán cubiertas en todo proyectos de desarrollo de software, lo cual permite garantizar que el producto resultante de este plan, tienen el nivel de calidad adecuado. Las secciones que contempla el plan de administración de la documentación de la IEEE son las siguientes.

### **2.6.1 PLAN DE ADMINISTRACION DEL PROYECTO CON EL ESTANDAR IEEE**

El plan de administración de proyectos documenta los asuntos relacionados con los requerimientos del cliente, los objetivos del proyecto, la organización del proyecto, la división del trabajo en tareas y la asignación de recursos y responsabilidades, y esta documentación esta referida en la norma IEEE Std. 1058.1-1993, la que consta de cinco secciones:

La primera sección del SPMP (SPMP: Software Plan Management Project, Plan de administración de proyectos), introducción, proporcionar la información de fondo par el resto del documento. Describe en forma breve el proyecto, los productos a entregar al cliente, los indicadores de avance del proyecto y los cambios que se espera que sufran los documentos. La sección contiene las restricciones fuertes que se encuentra en el acuerdo del proyecto y que son relevantes para los desarrolladores.<sup>28</sup>

La segunda sección del SPMP describe la organización del proyecto. Se describe el diseño de alto nivel del sistema, junto con los equipos de subsistemas y de funcionalidad cruzada que conforman el proyecto. Se definen las fronteras de cada equipo y de administración en la que se asignan responsabilidades. En está sección se describen los papeles de comunicación, como los enlaces. Leyendo esta sección los desarrolladores

---

<sup>28</sup> Bruegge, Dutoit, Ingeniería de software orientada a objetos. Prentice-Hall México 2002 Pag. 449



---

pueden identificar a los participantes en otros equipos con los que necesite comunicarse.

La tercera sección del SPMP, Proceso administrativo, describe la manera en que la administración supervisa el estado del proyecto y como ataca los problemas no previstos. Se describen las dependencias entre equipos y subsistema, y se hacen públicos los riesgos anticipados y planes de contingencia. Al documentar lo que puede salir mal se facilita que los desarrolladores lo reporten en cuanto sucedan los problemas. Esta sección deberá actualizarse con regularidad para que incluya los riesgos recién identificados.

La cuarta sección del SPMP, proceso técnico, describe los estándares técnicos que se requiere que se adopten todos los equipos. Estos van desde la metodología de desarrollo hasta la política de administración de la configuración de documentos y código, los lineamientos de codificación y la sección de componentes hechos estándar. Algunos pueden ser impuestos por intereses al nivel de la compañía y otros por el cliente.

La quinta sección del SPMP, elementos de trabajo, calendarización y presupuesto, representa el producto más visible de la administración. Esta sección detalla la manera en que se realiza el trabajo y quien deberá llevarlo a cabo. Las primeras versiones del SPMP contienen los planes detallados solo para las primeras fases del proyecto. Los planes detallados para las fases posteriores se actualizan conforme avanza el proyecto y se comprenden mejor los riesgos importantes.

Este plan de documentación es un estándar, que ha sido creado para regular los documentos emitidos en cada institución que desarrolla software. Esta norma permite asegurar que se esta realizando, un proceso de desarrollo de software que tiene las etapas necesarias para considera el producto resultante como de una calidad aceptable.

Esta documentación fue determinada por la IEEE y tiene como finalidad regular los procesos que deben seguirse para poder desarrollar software, se considera este plan de documentación como una herramienta a considerar para desarrollos de software. Ya que el dar seguimiento a este plan permite tener un producto aceptable según esta norma.



---

## **2.7 CONCLUSIONES.**

Toda actividad que es iniciada debe tener una preparación, esta preparación no excluye a los proyectos de software, pues en ella se determinan la serie de etapas que serán cubiertas durante el proyecto, estas etapas establecen un proceso a seguir, que es el punto de inicio en cada proyecto, asegurando proyecto bien planeado, tal como se plantea en la sección 2.1, el tener una preparación nos establece los lineamientos a cubrir durante el desarrollo de proyecto. Esto nos da como resultado el tener control de lo que realiza durante cada proyecto de software.

La preparación, como la definición de un proyecto pueden parecer actividades sencillas, pero es la parte de la formulación del proyecto que debe ser considerada como una de las de mayor importancia, ya que de una buena definición como de una buena preparación se desprenden una serie de resultados que den las condiciones propicias para realizar su administración. La definición de un proyecto evita que se tengan ambigüedades, puede garantizar un buen seguimiento durante el desarrollo del proyecto, logrando un software que cumpla con los requerimientos para los que fue planteado, tal como se estableció en la sección 2.2 referente a la definición del proyecto.

La formación de equipos de trabajo, es un tipo de organización para el desarrollo de software, este tipo de organización ha sido una de las que mayor utilización. Por lo tanto la formación de un equipo, que este integrado y tenga una comunicación adecuada hacia su interior, puede representar una ventaja en la administración de proyectos de software. La creación de equipos de trabajo, es una parte importante dentro de las actividades que se realizan en cada proyecto, tal como se pudo mencionar en la sección 2.3 referente a organización de proyecto donde el factor humano debe ser considerado, referente a base a sus habilidades y características que pueden considerarse en el personal que se desarrolle un proyecto de software.

Los costos son una parte importante de un proyecto, el saber identificarlos para realizar el presupuesto del proyecto, es una actividad que debe tener un cuidado especial ya que la estimación que pueda realizarse, determinara si el proyecto debe realizara o se cancelara, la estimación de costos es una de las etapas que no debe pasarse por alto ya que afecta directamente a su realización.



---

En un proyecto la comunicación es importante, ya que una buena comunicación facilita la generación de ideas, permite la de solución a problemas de manera conjunta. Una buena comunicación en un equipo de trabajo ayuda mantener una constante relación entre sus integrantes, tal como se menciona en la sección 2.4 referente a la importancia de comunicación en el equipo de trabajo, ya que las soluciones se pueden dar en el contexto de trabajar en forma conjunta y con una buena comunicación de ideas.

Parte importante de un proyecto es la documentación, pues es parte de la información que debe de generar un proyecto de software, pero esta documentación debe ser lo mas concisa y entendible en cada proyecto, en la sección 2.6 se presento un plan de administración de proyectos de software, que es resultado de un plan de documentación de proyectos de software, ya que de lo contrario cuando se carece de esta, se puede tener una desventaja administrativa, el tener un plan de documentación, tiene como finalidad el tener antecedentes de proyectos realizados y poder realizar estimaciones futuras a proyectos nuevos en base a estadísticas ya generadas de proyectos precedentes, este tipo de documentación debe determinar la organización que se siguió, el seguimiento realizado, modificaciones que puede haber tenido el proyecto, creando así un historial de los proyectos realizados y poder entonces manejar datos previamente establecidos, con los que puedan hacerse estimaciones precisas para proyectos fututos y poder elegir alternativas de solución mas precisas.



---

## **FUENTES DE CONSULTA**

Chiavenato Adalberto,  
Introducción a la teoría general de la administración.  
Mc Graw-Hill. Colombia.

Miklos Tomas. Tello Ma. Elena  
Planeación interactiva.  
Limusa. México. 2002

Miklos Tomas (Coordinador)  
Las decisiones políticas  
De la planeación a la acción.  
Instituto Federal Electoral. México. 2001

Robbins Stephen P. Coulter Mary,  
Administración  
Prentice Hall. Mexico. 2000. 6ta Ed.

Robbins, De Censo.  
Fundamentos de Administración.  
Prentice Hall, Mexico, 1999.

Romero Betancourt.  
Principios Fundamentales de Administración de Empresas  
Limusa. México, 2000.

Sommerville Ian,  
Ingeniería de software  
Adison Wesley 2002

[www.ieee.org/portal/index.jsp](http://www.ieee.org/portal/index.jsp)  
American Institute of Electrical Engineers.(IEEE).



---

### CAPITULO 3.

#### ELEMENTOS PROSPECTIVOS DEL DISEÑO.

Objetivo: Determinar el proceso de administración, que seguirán los proyectos de desarrollo de software, que proporcione elementos de solución a las problemáticas que presentan los proyectos de hoy.

La planeación de los recursos, la organización de equipos de trabajo, y la forma en que estos se puedan comunicarse, la documentación que cada proyecto genere, la programación de sus actividades, son tareas que tal vez no garantice el éxito de un proyecto, pero evitara que quede sin control y que sea cancelado. Los planteamientos expuestos en el capítulo previo, se consideran como parte del proceso de desarrollo de cada proyecto. Aunque hay procesos que ya se han establecidos resulta importante conocerlos.

Para administrar un proyecto de software, este debe tener un proceso administrativo que determine las acciones a seguir, en el capítulo previo se estableció una serie de acciones a realizar para poder iniciar y definir el proyecto, se planteo al equipo de desarrollo como principal elemento organizativo. Se determino la formación equipos de trabajo como forma de organización para el proyecto, se planteo la forma seleccionar a las personas, considerando sus atributos y características. Se dio especial importancia a los equipos de trabajo, ya que es una forma más comúnmente utilizada para trabajar en un proyecto y una vez terminado el equipo es desintegrado o pasa a formar parte de otro proyecto. El proceso administrativo entonces tiene definidas las siguientes funciones, como toda administración, dicha funciones son la definición del proyecto, la planeación del proyecto, la organización que se seguirá, y la forma en que se ejecutara y controlara.

Los procesos de desarrollo de software, hoy día son certificados internacionalmente, para asegurar que se cumple con las reglas de calidad del producto, el ciclo de vida del software, o la documentación que es generada en un proyecto, como en el caso del plan de documentación que se presento en el capítulo anterior, que fue definido por la IEEE. Los desarrollos de software continúan evolucionado constantemente, los cambios tecnológicos, las metodologías de desarrollo, los lenguajes de programación para desarrollar aplicaciones, y el surgimiento de nuevos procesos, requiere que las organizaciones dedicadas a desarrollar software, basen sus procesos en estándares internacionales, lo que les permite



garantizar que su producto cumple con estándares previamente establecidos. Uno de estos procesos que es utilizado para el desarrollo de software es, el Modelo de capacidad de Madurez, con el cual asegura que el procesos que se ha utilizado esta certificado por una institución que asegura que el proceso a sido llevado a cabo tal como lo marca la norma establecida.

Debido a que este es un proceso que es utilizando para el desarrollo de software, se considera que es importante conocerlo, ya que este puede ser una de los modelos que es considerado como alternativa de solución viable para la problemática planteada de los desarrollos de software. Por ello para tener un referente y una idea más clara de los procesos de certificación, se consideran dos de los más utilizados para el desarrollo de proyectos de software, estos son: el CMM y el ISO 12207.

### **3.1 EL MODELO DE CAPACIDAD DE MADUREZ (CMM)**

Debido a la problemática que presentan el desarrollo en organizaciones dedicadas a desarrollar software, se han establecido estándares que intentan normar el desarrollo de software. Una de estas instituciones que ha normado los procesos de desarrollo de software es, el Instituto de ingeniería de software (SEI), de la Universidad Carnegie-Melker instituto fundado por el Departamento de Defensa (DD) de los Estados Unidos de Norteamérica cuya misión es transferir tecnología de software. Este instituto estableció mejorar las capacidades de la industria de software en los EUA y, específicamente, las capacidades de aquellas organizaciones que reciben fondos del DD para proyectos de defensa. A mediados de los 80, el SEI inicio un estudio para evaluar a los contratistas que llevaban a cabo proyectos de software en el DD de los EUA.<sup>29</sup>

El producto del trabajo de valoración de la capacidad de la industria, fue el Modelo de la Madurez de Capacidades de Software del SEI. Tal modelo considera la mejora de procesos de desarrollo de software, este clasifica los procesos del software en cinco niveles diferentes. Estos cinco niveles se definieron por el instituto de la siguiente manera:

---

<sup>29</sup> Sommerville, Ingeniería de software, Addison Wesley, Mexico, 2002. Pag 568



#### **Nivel inicial.**

En este nivel, una organización no tiene procedimientos de administración efectiva o planes de proyectos. Aunque existen los elementos para el control del proyecto, no existen mecanismos organizacionales para asegurar que utilizan de forma consistente. La organización desarrolla software de forma exitosa pero las características del software (calidad) y los procesos del software (presupuesto, duración, etcétera) son impredecibles.

#### **Nivel repetible.**

La organización tiene procedimientos formales de administración, de afianzamiento de la seguridad y de control de la configuración. Se denomina repetible debido a que la organización puede repetir proyectos del mismo tipo de forma exitosa. Sin embargo, hay una falta de un modelo de procesos formal. El éxito del proyecto depende de los administradores individuales que motivan al equipo y de la organización que se pueda establecer en ese momento, y se actúa de forma intuitiva del proceso.

#### **Nivel definido.**

Para este nivel la organización tiene definidos sus procesos por lo que tiene una base para la mejora cualitativa de procesos. Existen procedimientos formales que aseguran que el proceso definido se sigue en todos los proyectos de software.

#### **Nivel administrado.**

Un cuarto nivel de la organización tiene un proceso definido y un programa formal de recolección de datos cualitativos. Recolecta las métricas del proceso y del producto para alimentar la actividad de mejora del proceso.

#### **Nivel optimizado.**

En este nivel, una organización esta comprometida con la mejora continua de procesos. La mejora de procesos se calcula y planea como parte integral de los procesos de la organización.

El siguiente grafico muestra los cinco niveles, por lo que pasa una organización para que logre el nivel optimo de desarrollo de procesos de software.



### Niveles de madurez, según el Instituto de Ingeniería del Software (SEI).



Figura 3.1

Los niveles de madurez en la versión inicial del modelo fueron señalados como demasiado imprecisos. Después de obtener experiencia al utilizar el modelo para la evaluación de la capacidad, se hizo una revisión en el año de 1993. Los cinco niveles se retuvieron pero se definieron más específicamente en términos de áreas de procesos.<sup>30</sup>

El trabajo del SEI en este modelo se vio influenciado por métodos de control estadístico de la calidad en la manufactura. Humphrey (1988), en la primera descripción publicada del modelo, señala que: W. E. Deming, en su trabajo con la industria japonesa después de la Segunda Guerra Mundial, aplicó los conceptos de control estadístico de procesos a la industria. Aunque existen diferencias importantes, estos conceptos son

<sup>30</sup> Sommerville, Ingeniería de software. Addison wesley Mexico, Pag 568



igualmente aplicables al software como lo son para los automóviles, las cámaras, los relojes de pulso y el acero. Factores como las habilidades y experiencia de los ingenieros de desarrollo afectan la calidad de los productos. A menudo, estos son tan importantes como los factores del proceso.<sup>31</sup>

Otro de los estándares que es utilizado en el desarrollo de software y que varias organizaciones han tomado para asegurar que sus productos cumplen los requerimientos de calidad, es el estándar creado por la ISO ((Organización Internacional de estándares), el cual esta basado en la norma para la administración de la calidad, Los estándares fueron establecidos para medir la calidad del producto. La norma ISO 12207, determina el plan que debe de cumplir la organización para alcanzar el nivel de certificación de los productos que deben de producir, así esta norma 12207 esta enfocada al plan de desarrollo que la organización, con la que se asegura que el producto que se espera cumple los estándares establecidos.

### **3.2 ESTANDAR ISO 12207.**

El estándar se basa en una serie de puntos que deben de cumplirse para que pueda ser valuado el proceso de calidad, el proceso se basa en el proceso organizativo del ciclo de vida. El cual es descrito en la norma en el apartado 7.1, la que determina varios puntos que deben seguirse, para poder cumplir con la norma.

#### **Proceso de Gestión de la norma ISO 12207.<sup>32</sup>**

El Proceso de Gestión contiene las actividades genéricas y tareas que pueden emplearse por cualquier tarea que tenga que gestionar sus respectivos procesos. El gerente es responsable de la gestión del producto, gestión del proyecto y gestión de las tareas de los procesos aplicables, tales como el de adquisición, suministro, desarrollo, operación, mantenimiento o soporte.

- 1) Inicio y definición del alcance.
- 2) Planificación.
- 3) Ejecución y control.

<sup>31</sup> Sommerville, Ingeniería de software. Addison wesley Mexico, Pag 570

<sup>32</sup> Norma internacional ISO/IEC 12207:1995.



- 4) Revisión y evaluación.
- 5) Terminación.

### **1. Inicio y definición del alcance.**

- 1.1. El proceso de gestión se deberá iniciar estableciendo los requisitos del proceso a emprender.
- 1.2. Una vez se han establecido los requisitos, el gerente deberá establecer la viabilidad del proceso comprobando que los recursos (personal, materiales, tecnología y entorno) requeridos para ejecutar y gestionar el proceso están disponibles, son adecuados y apropiados, y que los plazos para su terminación son alcanzables.
- 1.3. Tal como sea necesario, y por acuerdo de todas las partes interesadas, los requisitos del proceso pueden ser modificados en este momento para alcanzar los criterios de finalización.

### **2. Planificación.**

- 2.1 El gerente deberá preparar los planes para la ejecución del proceso. Los planes asociados con la ejecución del proceso deberán contener descripciones de las actividades y tareas asociadas, y la identificación de los productos software que serán proporcionados. Estos planes deberán incluir, sin estar limitados a ello, lo siguiente:
  - a) Plazos para la terminación a tiempo de las tareas.
  - b) Estimación del esfuerzo.
  - c) Recursos adecuados necesarios para ejecutar las tareas.
  - d) Asignación de tareas.
  - e) Asignación de responsabilidades.
  - f) Cuantificación de los riesgos asociados con las tareas o el mismo proceso.
  - g) Medidas para el control de calidad a emplear a lo largo del proceso.
  - h) Costes asociados con la ejecución del proceso.
  - i) Provisión del entorno e infraestructura.

### **3 Ejecución y control.**

- 3.1 El gerente deberá iniciar la implementación del plan para satisfacer los objetivos y criterios establecidos, ejerciendo control sobre el proceso.
- 3.2 El gerente deberá supervisar la ejecución del proceso, proporcionando informes internos del progreso del proceso e informes externos al adquirente tal como se defina en el contrato.
- 3.3 El gerente deberá investigar, analizar y solucionar los problemas descubiertos durante la ejecución del proceso. La solución de los problemas puede dar lugar a cambios en los planes. Es responsabilidad del gerente asegurar que se determina, controla y supervisa el impacto de cualquier cambio. Se deberá documentar los problemas y sus soluciones.



- 3.4 El gerente deberá informar, en momentos acordados, sobre el progreso del proceso, cumplimiento de los planes y soluciones a las situaciones de falta de progreso. Esto incluye informes tanto internos como externos, tal como requieran los procedimientos organizativos y el contrato.

#### **4 Revisión y evaluación.**

- 4.1 El gerente deberá asegurar que los productos software y los planes se evalúan con relación a la satisfacción de los requisitos.
- 4.2 El gerente deberá analizar los resultados de la evaluación de los productos software, actividades y tareas completadas durante la ejecución del proceso, con relación al cumplimiento de los objetivos y de los planes.

#### **5 Terminación.**

- 5.1 Cuando se completen todos los productos software, actividades y tareas, el gerente deberá determinar si el proceso se ha completado teniendo en cuenta los criterios especificados en el contrato, o como parte de un procedimiento de la organización.
- 5.2 El gerente deberá comprobar que los resultados y registros de los productos software actividades y tareas empleadas se han completado. Se deberán archivar estos resultados y registros en un entorno adecuado, tal como se especifique en el contrato.

Los modelos anteriores de desarrollo de software, como es el Modelo de Madurez del SEI (CMM), y el proceso de gestión de la norma ISO 12207, son procesos que se han adoptado para la realización de proyectos de desarrollo de software. Cada modelo de proceso determina sus propias actividades a desarrollar, establece sus lineamientos que seguirá el proceso, estos procesos son determinados según las condiciones que debe cumplir el proyecto, los modelos existentes han sido establecidos por instituciones, en el que el principal objetivo es normar los procesos. *Pero esto no asegura que todo lo que en ellos está establecido, de cómo resultado un proyecto eficiente, ya que son situaciones externas las que pueden llegar a afectar el proceso de cada proyecto.*



### **3.3 PROCESO ADMINISTRATIVO PROPUESTO PARA EL DESARROLLO DE SOFTWARE.**

La administración de proyectos es una de las ramas de la ciencia de la administración que trata del planteamiento y control de proyectos. Administrar un proyecto significa, en síntesis, planear su ejecución antes de iniciarla y vigilar dicha ejecución. El planteamiento del proyecto supone el establecimiento de metas y objetivos, la definición de las tareas que han de ser realizadas y la secuencia de dichas tareas en base a los recursos necesarios y disponibles.<sup>33</sup>

Basado en la definición anterior de administración de proyectos, para el diseño propuesto, estableceremos que todo proyecto de software consiste, de etapas a ser realizadas, definiendo un proceso, en que se define el objetivo del proyecto, su planeación, organización, ejecución y control, estas fases son las que se deben de cubrir para que pueda realizarse el proyecto, etapas que serán establecidas para lograr el objetivo de finalizar cada proyecto.

#### **3.3.1 DEFINICION DEL PROYECTO.**

Como ya se había mencionado en el capítulo previo, la definición del proyecto es parte del inicio de todo proyecto emprendido para ser desarrollado. Por lo que se retoma debido a la importancia que el inicio de la administración del proyecto.

Para iniciar un proyecto de software, se establecerán los objetivos que justificaran su inicio los resultados que de el se espera.

- Establecer la visión del proyecto
- La misión por cumplir y sus objetivos,
- La justificación del mismo, las restricciones y los supuestos.
- Las actividades a realizar.
- Los resultados se deben obtener,
- Los beneficios que se conseguirán.
- Los responsables del proyecto.

---

<sup>33</sup> Prado Darco, Administración de proyectos con PERT/CPM. Paraninfo Madrid 1988. Pag. 14



---

Esto permite establecer el alcance, y los objetivos que se perseguirán para el proyecto, se documentara cada proceso que se realice, para tener un documento informativo, que permita durante el ciclo del proyecto, hacer las correcciones pertinentes que resultasen necesarios realizar. Este tendrá los procesos que se seguirán durante su vida. Debe incluir el plan de desarrollo para poder determinar las actividades que tendrán que llevarse a cabo.

### **3.3.2 PLANEACIÓN DEL PROYECTO**

Deben determinarse los cursos de acción para lograr los objetivos, en la función de dicha planeación se definen las etapas, tareas y entregables (producto). Etapas en la que se divide el proyecto. Entregables en cada uno de las etapas, tareas a realizar (estructura de división del trabajo). Dentro del plan pueden ser definidos los planes para cada actividad que se realice, como es un plan técnico, un plan de negocio, o un plan de documentación. Para los proyectos de software es importante definir en el plan:

- Selección de ciclo de vida de desarrollo de software.
- Elaborar distintos tipos de planes (Administración de la configuración, control de la calidad, etc)
- Establecer relaciones y controles.
- Documentar el plan.
- Uso de estándares.
- Estimación del costo, duración del proyecto y recursos.

Al establecer el objetivo y realizar la definición del proyecto, acción seguida debe establecerse el plan de acción. Parte de la planeación es la de determinar el cronograma de actividades desarrollo de software, como de cualquier otro proyecto, es importante para poder manejar los recursos y poder en caso de ser necesario, realizar los ajustes pertinentes durante su desarrollo, esto permite tomar medidas de contingencia en casos de ser necesarios.



### **3.3.3 PROGRAMACION Y CONTROL DE PROYECTO.**

La programación de actividades, consiste en un conjunto de tareas que corresponden a una línea de tiempo. En la cual se puede revisar las definiciones hechas al plan del proyecto, y poder hacer los ajustes cuando cambian las restricciones o cuando se reasignan recursos.

Este calendario representa el trabajo desde el punto de vista del tiempo, contemplado dentro de las fechas que se ha propuesto tanto de inicio como de finalización de cada tarea en particular y de proyecto en forma general.

#### **PROGRAMACION DEL PROYECTO.**

La calendarización o programación inicial se crea estableciendo la correspondencia entre las tareas, el tiempo y los recursos. Las dependencias entre tareas, la duración estimada de las mismas, además se establecen la cantidad de personas que como participantes deben de ser utilizados para crear la calendarización inicial. En este se establecen las fechas de interacciones primarias con el cliente y/o los usuarios que estén involucrados en el proyecto, incluyendo las entrevistas, las revisiones del cliente y las entregas. Estas fechas se hacen parte del enunciado del problema que se presenta en el momento de realizar los acuerdos y representan fechas de entrega acordadas mutuamente entre el cliente y/o usuario y el gerente del proyecto. Las fechas se planean de tal forma que, aunque cambien, está satisfaga los tiempos de entrega.

La calendarización inicial debe incluir fechas de entrega para todos los productos que se entregan al cliente y una calendarización detallada a corto plazo. La programación detallada para el resto del proyecto se realiza conforme avanza el mismo. Además, una vez que se ha iniciado el proyecto cada equipo puede contribuir a la planeación de sus tareas, tomando en cuenta que conforme avanza el proyecto se tienen más información sobre el mismo. La programación se establece como un compromiso continuo entre recursos, tiempo y funcionalidad implementada, y se actualiza en forma constante.

Una forma grafica de representar las actividades programadas, es mediante los diagramas de Gantt. Desarrollado a principios del siglo XX. Por Henry Gantt. La esencia, es una grafica de barras en la que el tiempo esta representado en el eje horizontal y las



actividades a programar aparecen el eje vertical. Las barras muestran la producción, tanto planificada como el real, a lo largo de un periodo. El diagrama de Gantt<sup>34</sup> muestra visualmente cuando se ha supuesto, que deberán realizarse las tareas, y compara esa situación ideal con el avance real lograda en cada una de las fases. Lo cual permite visualizar el avance del proyecto y tomar las medidas correctivas necesarias para lograr el objetivo del proyecto. (Grafica 3.1)

### DIAGRAMA DE GANTT<sup>35</sup>

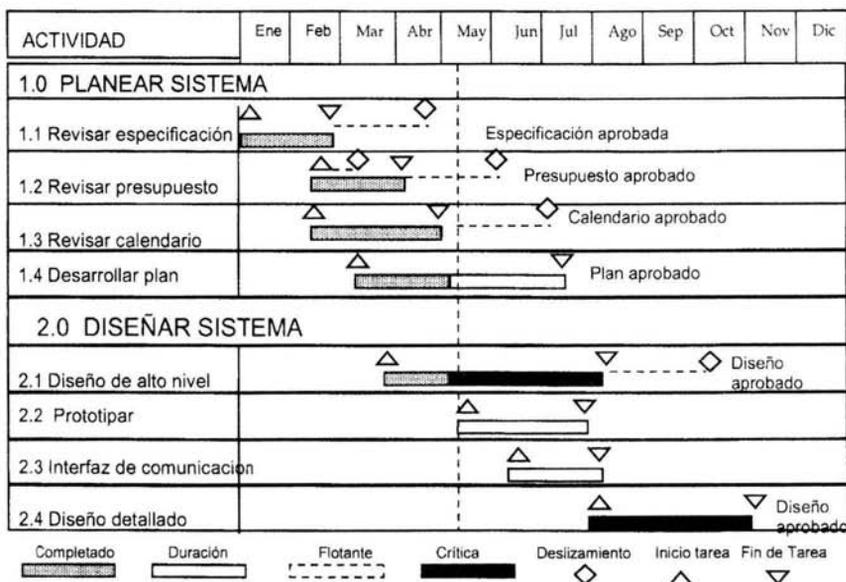


Grafico 3.1

<sup>34</sup> Robbins, Administración, Mc graw Hill, Mexico 2002. Pag 250

<sup>35</sup> Pileerger Shari Lawrence. "Ingeniería de software" un enfoque practico, Prentice-Hall México, Pag 101



### 3.3.4 ASIGNACION DE RESPONSABILIDADES

La administración de proyectos asigna dos tipos de papeles, uno es a los desarrolladores a quienes se les puede asignar responsabilidades de administrativas, como de líder de equipo y enlace, y papeles técnicos, como analista o desarrollador, la administración asigna estos papeles a un individuo o a un equipo, dependiendo de los requerimientos de esfuerzo y el tiempo asignado.

La asignación de papeles administrativos se asigna a participantes individuales. Un papel como el de líder de equipo, no puede desempeñarse de manera efectiva si esta compartido por dos o mas participantes. Un líder de equipo cubre una función de comunicación actuando como enlace entre los altos niveles de administración y los desarrolladores, por tanto debe poder comunicar una imagen consistente del estado del proyecto en ambos sentidos. Segundo, aunque un líder de equipo toma decisiones buscando por lo general, el consenso de desarrolladores, en ocasiones debe de imponer decisiones críticas de tiempo. Otros papeles administrativos, sirven de enlace entre equipos de desarrollo, y requieren un mismo grado de consistencia y esta actividad debe de ser asignada a un solo individuo.

La administración debe de buscar tres cualidades cuando asigna papeles administrativos:

- La capacidad para comunicarse,
- la capacidad de reconocer pronto los riesgos y
- la disciplina para separar las decisiones administrativas técnicas.

Tales cualidades provienen de las experiencias en el establecimiento de proyectos.

La productividad de un quipo de trabajo, debe tener las herramientas que son necesarias para el desarrollo de sus funciones, estudios realizados se ha comprobado que, las personas cuando son provistas de un entorno de trabajo propicio para desarrollar sus actividades, logran incrementar su productividad. Por eso de considera importante establecer el entorno de trabajo.



### 3.3.5 ENTORNO DE TRABAJO

El lugar de trabajo tiene efectos importantes en el desempeño de las personas y en su satisfacción del mismo, con el que el producto puede ser , y el . Experimentos psicológicos han mostrado que el comportamiento se ve afectado por el tamaño de la habitación, el mobiliario, el equipo, la temperatura, la humedad, la brillantez de la calidad de la luz, el ruido y el grado de privacidad disponible. El comportamiento del grupo se ve afectado por la organización arquitectónica y los recursos de comunicación. Las comunicaciones dentro del grupo se ven afectada por la arquitectura del edificio y la organización del lugar del trabajo.

A menudo, el personal de desarrollo de software trabaja en áreas y oficinas abiertas, algunas veces en cubículos, y solo los administradores principales tienen oficinas individuales. McCue llevo a cabo un estudio que muestra que la arquitectura abierta que favorecía a muchas organizaciones ni era popular ni productiva. Los entornos más importantes que se identificaron en ese estudio de diseño fueron:

- a) **Privacidad.** Los programadores requieren un área donde se puedan concentrar y trabajar sin interrupción.
- b) **Conciencia del exterior.** Las personas prefieren trabajar con luz natural y con vista al exterior.
- c) **Personalización.** Los individuos adoptan diferentes prácticas de trabajo y tienen diferentes opiniones sobre la decoración. La habilidad para arreglar el lugar de trabajo que permita llevar a cabo éste y personalizar ese entorno es importante.

Las personas les gustan las oficinas individuales que puedan organizarlas como ellas quieran. Existen menos distracciones y pocas interrupciones que en espacios de trabajo abiertos. El entorno de trabajo puede ser parte fundamental de lograr que un equipo de desarrollo logre la finalización de los proyectos con un buen nivel de comunicación y de desarrollo profesional.



### 3.3.6 DIRECCION

La realización de planes implica la ejecución de actividades, una dirección eficaz significa impulsar y conducir los diferentes grupos humanos, hacia la finalización de los objetivos. En la administración de proyectos debe determinar quien hará y quien dirigirá las acciones a seguir en el proyecto.

En esta función de la administración, la dirección del proyecto determina quien realizara la ejecución del plan, para ello existe una figura importante que es la de jefes de equipo.

#### LOS JEFES DE EQUIPO.

La administración de un proyecto es una actividad intensamente humana, por esta razón algunos proyectos no logran desarrollarse completamente en equipo, y no por la falta de capacidad de las personas, sino que no se ha hecho la combinación adecuada de las capacidades humanas del personal para poder dirigir a las a personas.

Así quien dirija un equipo de trabajo, debe tener seguir el modelo propuesto por Jerry Weinberg quien, intenta responder a esta pregunta sugiriendo el modelo de MOI (Motivación, Organización, Ideas e Innovación):<sup>36</sup>

**Motivación.** La habilidad para motivar (con un «tira y afloja») al personal técnico para que produzca conforme a sus mejores capacidades.

**Organización.** La habilidad para moldear procesos existentes (o inventar nuevos) que permita al concepto inicial transformarse en un proyecto.

**Ideas o innovación.** La habilidad para motivar al personal para crear y sentirse creativos incluso cuando deban de trabajar dentro de los límites permitidos para un producto o aplicación de software particular.

Weinberg, sugiere que el éxito de los administradores de proyecto se basa en aplicar un estilo de administración en la resolución de problemas. Es decir, un gestor de proyecto de software debería concentrarse en entender el problema que hay que resolver, gestionando el flujo de ideas y, al mismo tiempo, haciendo saber a todos los miembros del

---

<sup>36</sup> Range S Pressman. Ingeniería de Software. Un enfoque practico, Mc graw-Hill Mexico 2000. Pag 180



equipo (mediante palabras, y mucho más importante, con hechos) que la calidad es importante y que no debe verse comprometida.

Los equipos no son recursos automáticos para elevar la productividad. Además es posible que sean una decepción para la gerencia. Se debe examinar con más detenimiento la forma en como los gerente pueden desarrollar y administrar equipos eficaces y como se utilizan dichos equipos en iniciativas de control. Sin embargo, veamos antes cuales son las características de un equipo eficaz.

### **CARACTERÍSTICAS DE LOS EQUIPOS EFICACES**

Las investigaciones recientes aporta conocimientos sobre las características principales que se asocian con los equipos eficaces, este tiene que ver con la creatividad de cada una de las personas que pueden ser empleadas en la formación de un equipo. Dichos equipos deben de definir las metas y buscar las habilidades que pueden hacer que el equipo de trabajo pueda retroalimentarse para poder tener un desarrollo satisfactorio en cada etapa del proyecto que se este iniciando. Para comprender esto puntos de los equipos que resultan ser parte de la dirección de los equipos de trabajo.

**Metas claras.** Los equipos de alto rendimiento deben de tener una clara comprensión de la meta por alcanzar y una firme convicción de que el logro de dicha meta será resultado valioso o importante. Más aun, la importancia de tales metas alienta a los individuos a reorientar sus inquietudes personales hacia esos objetivos colectivos. En los equipos eficaces, los miembros se comprometen a alcanzar las metas colectivas, saben que los logros se esperan de ellos y entienden como pueden trabajar juntos para alcanzar esas metas.

**Habilidades pertinentes.** Los equipos eficaces están formados por individuos competentes. Ellos poseen las habilidades técnicas y las destrezas necesarias para alcanzar las metas deseadas, y cuentan con las características personales requeridas para alcanzar la excelencia mientras trabajan satisfactoriamente con otras personas. Un segundo elemento es importante y a menudo se pasa por alto, el que todas las personas técnicamente competentes poseen las habilidades necesarias para trabajar bien como miembros de un equipo. Los equipos de alto rendimiento están constituidos por individuos dotados de habilidades tanto técnicas, como habilidades para relaciones personales.



### 3.4 ESCENARIOS.

Nuestros sucesos futuros, dependen sobre los cambios que podamos hacer hoy. Estos cambios, dependen del entendimiento y suposiciones acerca del futuro que nos puede traer.<sup>37</sup>

Los escenarios son parte de la planeación estratégica que permite que se desarrollen, supuestos entendibles y plausibles, para ello deben determinarse las variables que podemos estar calificando y la forma en que estas pueden vincularse, para un manejo prospectivo de la forma en que este pudiera ocurrir en un determinado tiempo. Los escenarios solo serán comprobados llegando el tiempo para el que se haya planteado.

El proceso de creación de escenarios pone gran énfasis sobre las articuladas definiciones de una problemática y sobre la síntesis de ideas, en lugar de solo extender y profundizar en un análisis de un solo punto de vista. Estos involucran el uso de múltiples perspectivas para explorar problemas, los escenarios nos ayudan a crear un entendimiento compartido de posibles desarrollos, opciones y acciones.

La formulación de escenarios, permite a una organización estar preparada para los posibles sucesos que se puedan presentar y afrontarlos en condiciones favorables, cambios que necesariamente se dan en cualquier ámbito organizacional.

Para el presente trabajo se ha propuesto la creación de un proceso de administración de proyectos, que da elementos de solución para la problemática planteada. Se ha propuesto un sistema de administración, basado en la administración de proyectos como herramienta que ayude a solucionar las carencias que se presenta en cada proyecto que es iniciado. En la formulación de escenarios se considera que para plantearlos debe especificarse las variables sobre las que se basaran los escenarios. Por las condiciones para las que se ha desarrollado la presente investigación, se propone como primer variable principal a la "Gestión Administrativa", definida por el proceso de desarrollo de software, conformada por la definición de proyecto, la planeación, organización y control.

La formulación de escenarios se requiere de por lo menos dos variables principales, para poder definir un planteamiento prospectivo, que determine los posibles futuros, del escenario planteado. La segunda variable principal a considerar en la formulación de

---

<sup>37</sup> Ged Davis, "Scenarios as Tool for the Century XXI", Julio 12 de 2002. Pag 1



---

escenarios es, la normatividad a seguir por organizaciones que regulan los procesos de gestión para el desarrollo de software.

Estas variables consideras como las principales para la formulación de escenarios, deben poder ser medidas, estos valores pueden ser cuantitativos como cualitativos, por lo que para la gestión administrativa se considera un valor de tipo cualitativo, dicho valor será el de eficiente, si sigue el proceso propuesto con las condiciones establecidas por el proceso administrativo, y otro valor que será el deficiente si no cumple con el proceso establecido o no tiene el proceso de realización de proyectos.

Las variables primarias determinan el planteamiento des sistema que se esta proponiendo, y para formular el escenario debe también determinarse las variables secundarias, que forman parte del entorno del sistema que se esta proponiendo, las variables secundarias que intervienen en el planteamiento de los escenarios y consideradas para esta investigación son las siguientes:

La primera variable secundaria es la tecnología, definida como las metodológicas de desarrollo de software, el hardware que se requiere para el desarrollo, para el que se considera una escala de medición de la siguiente manera: obsoleta, actualizada y de vanguardia,

La capacitación para el personal que es integrado a un equipo de trabajo para la realización de un proyecto, donde la capacitación esta basada en los atributos y las habilidades que tenga el personal del equipo de trabajo. Con el se pueda lograr tener un personal calificado y orientado a proyectos de desarrollo de software.

Para realizar el escenario, se propone una última variable secundaria, esta última variable a considerar es la documentación que representa un medio de información que sirve como retroalimentación para futuros proyectos. Esta documentación servirá para futuros proyectos, lo que permitirá tomar decisiones más acertadas, pues ya se tiene registro de los proyectos realizados.

Por lo que resumiendo tenemos las siguientes variables propuestas y su escala de medición.



### 3.4.1 VARIABLES PRINCIPALES:

#### Gestión administrativa.

Definida como el proceso de administración, para el desarrollo de proyectos de software, en el que se contemplan las funciones administrativas propias para desarrollo de software. Como es la definición de proyecto, la planeación, la organización, el control y dirección del proyecto.

#### Normatividad.

Definida como la serie de reglas, para alcanzar dentro del estándar de CMM el nivel de procesos adecuado para un nivel de calidad de software, para el que se han definido los niveles de proceso para la elaboración de software, considerado como repetible, definido, administrado y optimizado.

La escala de medición para las variables propuestas, se han considerado por una escala propia, ya que uno de los principales problemas de los proyectos de desarrollo de software es, la carencia de datos estadísticos de cada proyecto realizado. Por ello se propone la siguiente escala de medición.

#### ESCALAS DE MEDICIÓN DE LAS VARIABLES PRIMARIAS.

Nombre de variable	Escala de medición	Valores que puede tomar
Gestión Administrativa	Se siguen procesos definidos, administrativos, de desarrollo de documentación, de control es decir se tiene el proceso bien establecido. No se siguen los procesos establecidos	Eficiente.  Deficiente
Normatividad	<b>Nivel Repetible.</b> Se considera la realización de software, sin tener metodologías de desarrollo y carentes de documentación. <b>Nivel Definido.</b> Se cuenta con registros de proyectos previos, que permiten retroalimentación <b>Nivel Administrado.</b> Se cuenta con procesos definidos y existen planes para cada actividad que se realiza. <b>Nivel Optimizado.</b> Se tiene un proceso de mejora continua, y los proyectos están basados en proceso normados.	1 a 2  3 a 4  5 a 7  8 a 10

Cuadro 3.1



### 3.4.2 VARIABLES SECUNDARIAS:

#### Tecnología.

Definida para el escenario como, la herramienta necesaria para desarrollar proyectos de software, en el que se considera el hardware, el software como herramienta de apoyo para generar otro software. Las metodológicas de desarrollo que sirven de apoyo para la elaboración de software, además de lenguajes de programación, herramientas de diseño de programas. Su escala de medición es: Obsoleta, Eficiente, De vanguardia.

#### Capacitación.

Capacitación con base en atributos y características de las personas, que están en el proyecto o que se integran a proyectos, la capacitación debe ser de manera constante, ya que la creación de software, debe considerar por su naturaleza, los elementos tecnológicos de vanguardia. Esta debe ser continua, basada en lo atributos y características que tenga el personal.

#### Documentación.

Definida como sistema de información de registro de proyectos realizados o en desarrollo, que sirvan de retroalimentación de para futuros proyectos, para tomar las decisiones con base a una base de datos confiable y oportuna, permitiendo hacer los ajustes necesarios a los sistemas propuestos. (Ver cuadro 3.2)

#### ESCALAS DE MEDICIÓN DE LAS VARIABLES SECUNDARIAS.

Nombre de variable	Escala de medición	Valores que puede tomar
Tecnología	<p><b>De vanguardia.</b> Se cuenta con un plan de renovación tecnológica, se actualizan equipos de cómputo, y se exploran nuevas herramientas de desarrollo.</p> <p><b>Suficiente.</b> La tecnología no se cambia constantemente, pero permite seguir operando y realizando desarrollos de sin realizar cambios inmediatos.</p> <p><b>Obsoleta.</b> La tecnología no permite realizar desarrollos, la interrelación con otros sistemas ya no es operable, y tiende a morir</p>	<p>8 a 10</p> <p>5 a 7</p> <p>0 a 4</p>



Capacitación	Se tienen un programa de capacitación para el personal de acuerdo a las características del personal.	9 a 10
	Capacitación continua pero no enfocada a atributos y habilidades de personal.	6 a 8
	Capacitación diferente para las habilidades que tiene el personal y que no están dentro de una capacitación de desarrollo personal.	1 a 5
	No hay capacitación	0
Documentación	Documentación que es eficiente, clara y concisa disponible en todo momento, que permite tomar información para la toma de decisiones.	8 a 10
	Documentación suficiente pero carente de eficiencia, que requiere de un análisis para su entendimiento	5 a 7
	Documentación insuficiente que no tiene elementos para poder ser considerada como información con certeza.	1 a 4
	Carente de documentación.	0

Cuadro 3.2

Habiendo definido la escala de medición para cada variable, tanto de las variables primarias como de las secundarias, se está en posibilidad de plantear los escenarios a futuro, para realizar el análisis prospectivo respecto a las variables previamente definidas para el desarrollo de software, podemos plantear escenarios plausibles de realizarse, esta prospectiva se realiza para los años 2005, 2010 y 2015. Para poder visualizar las variables, estas las colocaremos en un plano, en la que se establecerán las variables definidas previamente. Y realizar las posibles tendencias que se presentaran dependiendo del manejo se pueda realizarse de estas.

### 3.4.3 FORMULACIÓN DE ESCENARIOS.

Los escenarios planteados se han denominado como Óptimo, Ideal, Constante y de Oportunidad. Dichos escenarios respecto a tiempo de realización planteados en términos de realización de la Gestión de proyectos que se pueda ir realizando será la siguiente:



## ESCENARIO PLANTEADO PARA EL AÑO 2005

*"Las gestión empieza a dar resultados"*

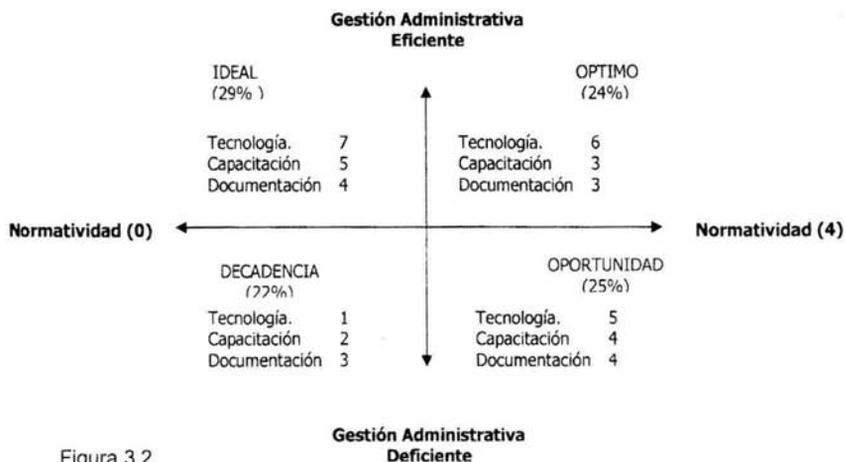


Figura 3.2

La prospectiva planteada para el año 2005, nombrada la "gestión que empieza a dar resultados", las actividades ya tienen un orden de realización, la probabilidad de ocurrencia para cada escenario es la siguiente, 29% para el escenario ideal, 24% para el escenario óptimo, 22% para el de decadencia, y un 25% para el escenario Oportunidad. (Figura. 3.2)

Para el año 2005 se espera, que el escenario ideal tenga mayor probabilidad de realización, pues se tienen elementos que permiten empezar a vislumbrar una mejoría, ya que la tecnología es suficiente, se inicia con documentación de proyectos que son realizados, documentación todavía insuficiente para realizar un diagnóstico pero ya se tiene un buen inicio. La capacitación se inicia, aunque todavía no se realiza con respecto a las características y habilidades del personal. El escenario óptimo aun no logra tener las condiciones necesarias para ser el de mayor probabilidad, se inicia un proceso de desarrollo de proyectos, pero todavía se está en el nivel más bajo de la normatividad es decir en el que las actividades son realizadas de manera repetitiva sin ser una forma sistematizada. El escenario de Decadencia todavía está latente en la organización. El escenario Oportunidad hace que todavía los desarrollos sean finalizados aun y en condiciones que pudiesen presentarse adversas.



### ESCENARIO PLANTEADO PARA EL 2010.

"Ya mejoramos, pero queremos hacerlo mejor"

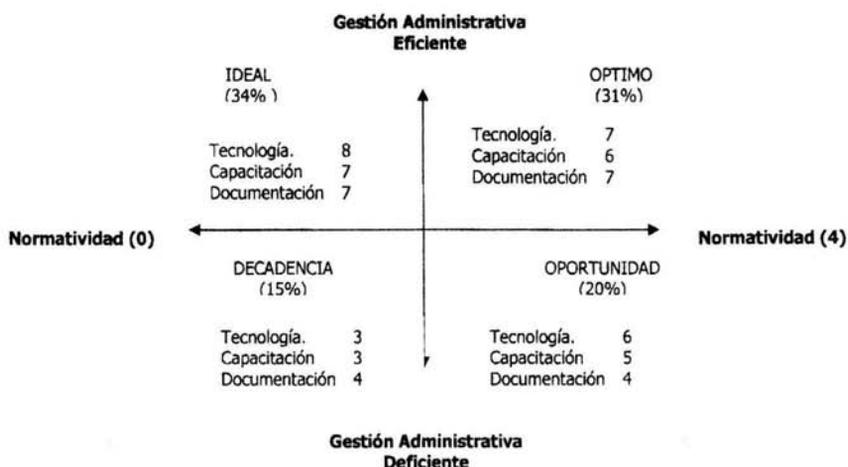


Figura 3.3

Los escenarios planteados para la el año 2010, se presenta con mayor certeza el escenario ideal, ya que se cuenta con tecnología de vanguardia, y el personal tiene capacitación continua apegada a los atributos de cada persona. La documentación, ya es suficiente para poder realizar diagnósticos más certeros de desarrollo de proyectos y poder realizar con la información registrada, las decisiones basadas en información mas acertadas. La gestión administrativa debe ser ya un proceso sistémico eficiente. Los escenarios planteados se pronostica tenga las probabilidades de 34%, 31%, 15% y 20% para los escenarios Ideal, Óptimo, Decadencia y Oportunidad respectivamente. Para estos escenarios el nivel de normatividad es el de un nivel de administrado, ya que se tienen las bases de la gestión, la cual debe ser capaz de aprender y adaptarse para las condiciones futuras. El escenario de decadencia dejara de ser una amenaza latente en la organización y el escenario de Oportunidad debe cambiar a un escenario ideal, para empezar ha hacer las cosas de forma optima. (Figura 3.3)



### ESCENARIO PLANTEADO PARA EL 2015.

"Debemos llegar al ideal y poder mantenerlo"

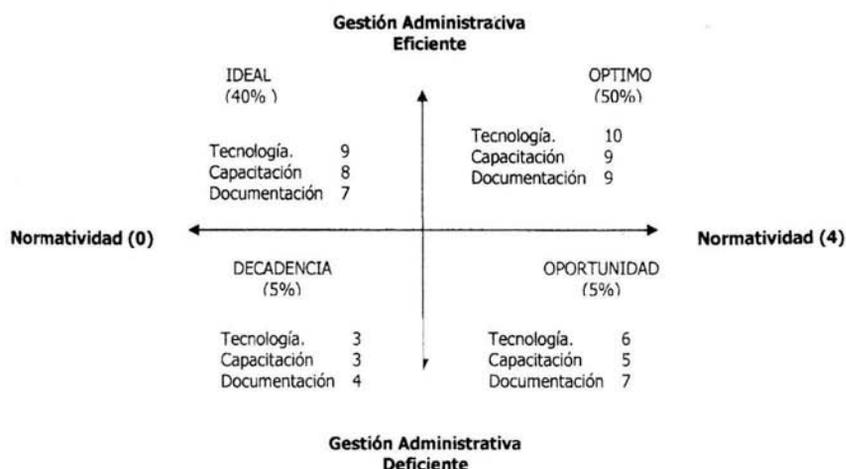


Figura 3.4

Para el año 2015, se espera que el escenario óptimo ya sea una realidad, debe tener una probabilidad mayor de realización, ya que el escenario ideal ha sido alcanzado, y las condiciones son propicias para alcanzar un nivel óptimo. Para que este escenario este dentro de los parámetros de probabilidad de realización, se considera que la tecnología que se esta utilizando sea de vanguardia, el personal integrado en un proyecto de desarrollo tiene un programa de capacitación de acuerdo a las habilidades y características del personal y del tipo de proyecto que desarrolle., la documentación es oportuna y clara, lo que permite tener una retroalimentación de proyectos, el escenario Oportunidad tendrá que pasar necesariamente al escenario ideal. El escenario de decadencia tiende a desaparecer, lo que representa una amenaza a las organizaciones. Las probabilidades de ocurrencia quedan de la siguiente manera: Ideal 40%, Óptimo 50%, en decadencia 5%, y el escenario de oportunidad será de 5%. Este escenario prospectivo será el de mayor fortaleza, para el desarrollo de proyectos, con ello el nivel de normatividad debe lograrse hasta el nivel de Óptimo y asegure un proceso eficiente de desarrollo de administración de proyectos.



### 3.5 CONCLUSIONES

En el presente capítulo se presentaron estándares de procesos que son utilizados hoy día en la realización de software, uno de ellos es el Modelo de Capacidad de Madurez del Instituto de ingeniería de software y el estándar 12207 de la ISO (Organización Internacional de Estándares) que es utilizado para la gestión reproyectos. Estos estándares expuestos en las secciones 3.1 y 3.2 determinar la normatividad para los desarrollos de software, debido a que un producto de software que es utilizado, debe de tener sus procedimientos bien establecidos para poder competir en un entorno globalizado. Así el software que se realiza tendrá que normar sus procedimientos y crear sus metodologías que le apoyen a entrar en un mercado competitivo, aunque en esta investigación se planteo para desarrollos a la medida, esto no implica que un proyecto exitoso en un futuro próximo pueda llegar a comercializarse.

La existencia de procesos normativos de gestión, no debe ser una limitante para crear nuevos sistemas de gestión que mejoren los ya existentes, por lo que en la sección 3.3 se planteo un proceso de administración para desarrollo de software, en el que se plantearon las principales actividades a realizar en un proyecto de software, en el que se plantearon las principales actividades a realizar en cada proyecto, que pueda ser controlado y dirigido, en donde los diagramas de Gantt, siguen y seguirán siendo una herramienta de apoyo para los proyectos.

La formación de equipos es una forma de organización para el desarrollo de proyectos, se plantea la formación de estos equipos, ya que la conformación de un equipo de trabajo eficiente puede obtener el máximo beneficio de las habilidades y destrezas de sus integrantes, que con un ambiente de trabajo adecuado puede logra tener una buena productividad como se menciona en la sección 3.3.5.

Los procesos administrativos, se realizan en función de las actividades que deben de realizarse en cada proyecto, como es el planear, organizar, controlar y dirigir. En la sección 3.3 se propone el proceso de administración de proyectos de software, proceso que tiene como objetivo principal el lograr que los proyectos sean finalizados de manera satisfactoria, que se logren cubrir las etapas propuestas, siguiendo el proceso administrativo apropiada para el desarrollo de software. Con el que se espera poder solucionar las principales deficiencias que tienen los proyectos de software.



Con las condiciones del proceso administrativo se persigue un ideal, que debe ser puesto en práctica e ir dando solución a la problemática planteada, en tal proceso debe haber un registro de lo realizado, creando una documentación del proyecto, para con ello tener una base de información que genere que de apoyo para cada nuevo proyecto y este proceso administrativo pueda ser valuado, solo después de que se realicen las actividades propuestas. El proceso que se propone no es único y es perfectible, pero a decir de Ackoff, este deberá ser un sistema que se sea capaz de aprender y de adaptarse.

El entorno en el que se desarrolla software, es cambiante y parte de estos cambios provocan condiciones diferentes en el desarrollo de software con respecto al tiempo, por que debe de plantearse escenarios de cómo puede ser los proyectos en un futuro próximo. Así en la sección 3.4 se han planteado escenarios plausibles de realizarse, que han sido planteados sobre una serie de variables que afectar el entorno de cómo se presenta cada proyecto. Estos escenarios determinan una visión a futuro según las condiciones establecidas, donde los escenarios deben de pasar de un estado ideal a un estado optimo y que la realización de las tareas planteadas determinaran el grado de madurez que alcance la administración planteada, los que serán valuados por los estándares de normatividad planteados en la sección 3.1 del CMM (Modelo de Capacidad de Madurez), además de tener en consideración que se han establecido las bases para poder cubrir condiciones propicias para lograr finalizar los proyectos con una gestión administrativa, que incremente el nivel de finalización de proyectos que son iniciados, y lograr un nivel de CMM optimo para los procesos de desarrollo de software, que pueda cumplir el nivel de calidad que establece la Organización Internacional de Estándares.

El software sigue y seguirá siendo una tecnología con rápido crecimiento, que crea sus propias herramientas de desarrollo, con el se realizaran investigaciones en todas las áreas de conocimiento, porque hoy es una herramienta que agiliza muchas actividades económicas, financieras, medicas entre otras. Pero su desarrollo seguirá dependiendo del manejo que el hombre pueda hacer de este, pues de el depende toda la creatividad que pueda tener para generar cada vez software con aplicaciones mas amplias, y determinara las forma y procedimientos que sean adecuados para realizar esta gran tarea de creatividad. Por que el software seguirá siendo una actividad creativa, que únicamente necesita ser administrada de forma eficiente y efectiva para generar nuevo conocimiento.



---

## **FUENTES DE CONSULTA**

Bruegge Bernd, Dutoit Allen H.  
Ingeniería de software orientada a objetos  
Prentice Hall. México. 2002

Chiavenato Adalberto  
Introducción a la teoría general de la administración.  
Mc Graw-Hill. Colombia

Domingo Ajenjo Alberto,  
Dirección y Gestión de Proyectos  
Un enfoque practico,  
Alfaomega Ra-Ma. Mexico. 2000

Gido Jack y Clements James P.  
Administración exitosa de proyectos  
Internacional Thomson Editores. Mexico. 1999

Hernández Jiménez Ricardo  
Administración de la función informática  
FACTOR AFI, Trillas. Mexico. 2003

Van Der Heijden Kess  
Escenarios  
El arte de prevenir el futuro.  
Panorama. México. 2000.

[www.iso.org/iso/en/ISOOnline.frontpage](http://www.iso.org/iso/en/ISOOnline.frontpage)  
Organización internacional de estándares Internacionales (ISO).

[www.ieee.org/portal/index.jsp](http://www.ieee.org/portal/index.jsp)  
American Institute of Electrical Engineers.(IEEE).

[www.software.net.mx/inicio/default.htm](http://www.software.net.mx/inicio/default.htm)  
Sitio oficial de la industria mexicana de software.



---

## CONCLUSIONES GENERALES

El software es una de las tecnologías que mayor crecimiento ha tenido en los últimos tiempos, su desarrollo esta en constante cambio, su evolución es rápida, que durante estos constantes cambios se han presentado una serie de problemáticas, que deben verse como una serie de oportunidades, para proponer soluciones a esta serie de eventualidades que se generan, muchos pueden ser los problemas que se pueden presentar, pero siempre existirá la capacidad de poderlos afrontar y solucionar, ya sea creando los medios para solucionarlo o buscando herramientas necesarias para realizar proyectos. Por lo que podemos concluir del primer capítulo, que el desarrollo de software seguirá evolucionando, planteando nuevos retos, pero especialmente representa un reto del orden administrativo, que cumpla con los objetivos para los que se plantea un proyecto de software. Que la problemática que se presenta hoy en día, tiene una serie de causas de orden administrativo, provocando que los recursos estén desaprovechados y los costos sean incrementados. Es por ello que resulta importante saber administrar proyectos de software, saber detectar las causas que la provocan la problemática y la forma que deben ser resueltas, así del diagnostico que se realizo en el primer capítulo desde la evolución del software hasta la formulación de la problemática, nos indica que deben de buscarse soluciones creativas factibles de realizarse Gran parte de las organizaciones que se dedican al desarrollo de software presentan esta problemática, por lo que puede establecerse que esto no es un problema de una única organización, si no que representa una de las mayores complejidades que se enfrentan hoy día el desarrollo de software y con ello la administración de estos proyectos.

La administración de proyectos de software, es una actividad que presenta una serie oportunidades de innovaciones, además de un desarrollo profesional muy amplio, pero para ello habrá que realizar y proponer procesos administrativos que den solución a la problemática descrita, tal como se plantea en el capítulo dos, donde se establecen una serie de puntos a considerar en cada proyecto de software, que aseguran que los proyectos sean realizados de tal manera que el proyecto sea concluido dentro de los tiempos y presupuestos previamente establecidos para determinado proyecto, objetivo que se cumplirá con las actividades previamente establecidas durante los procedimientos que debe cubrir un proyecto.



Las actividades que se establecieron de una serie de actividades que pudieran parecer obvias, son las que muchas veces se prescinde de ellas, está serie de puntos como la preparación del proyecto, la definición, la asignación de recursos son parte fundamental para la realización de un proyecto de software, ya que son parte medular para poder iniciar un proyecto, pero si se carece de estas elementales actividades estaremos siempre con una debilidad administrativa.

El Personal que se ve relacionado con el proyecto, debe seleccionarse dependiendo sus habilidades y atributos que tengan las personas que formen un equipo de trabajo. Por ello podemos decir que todo proyecto que se realice deberá considerar los elementos fundamentales que se establecen en el capítulo dos y debe considerar el factor humano como parte fundamental de la realización de un proyecto de software, ya que sin este se carece sentido de poder desarrollar software.

Las normas planteadas de CMM y la ISO 12207 que se mencionaron en la presente investigación, son una muestra de que se están realizando investigaciones, que tratan de solucionar de alguna manera el mismo tipo de problemática planteado en el presente trabajo. Pero el que existan estas norma, nos puede servir de referente y considerar si las adoptamos o no, ya que cada problema debe ser analizado para generar sus propias soluciones, por lo que es importante conocerlas estándares y determinar de qué manera nos pueden ayudar o definitivamente no considerarlas.

En el capítulo último se realizó la propuesta de proceso de administración, que establece los pasos a seguir durante el desarrollo de software, tal proceso tiene como principal elemento el factor humano, que en base a determinar sus capacidades y actitudes del personal, podrán formarse equipos creativos para el desarrollo de proyectos. Y con ellos incrementar el número de proyectos finalizados.

La formulación de escenarios, plantea tres futuros probables de ocurrir si se realiza una gestión administrativa, de acuerdo a las condiciones y los lineamientos establecidos en la propuesta realizada. La cual no puede ser valuada de forma inmediata, ya que requiere de su puesta en práctica, pero que se puede establecer un nivel comparativo con respecto del CMM, con el que se espera llegar a un nivel de óptimo del CMM. Con lo que puede esperar que los proyectos cumplen con elementos necesarios para ser considerado en un nivel de madures del CMM.



---

La administración necesariamente la realizamos en algún momento, las personas que realizan actividades de programación, desarrollo de programas, realizan actividades de un nivel técnico, pero en algún momento se tiene que realizar actividades de tipo administrativo, por lo que esta investigación establece la base de un buen principio para la administración de proyectos.

La administración de proyectos de software, es una actividad de la administración que esta en constante evolución, ya que el software que hoy utilizamos el día de mañana puede ser obsoleto. Este constante cambio, fue pauta de inicio del presente trabajo, ya que se planteaba una falta de administrar proyectos, en el que se carece de una planeación y que debido a ello los proyectos tienen indefiniciones, no se respetan tiempos estimados y al pasar del desarrollo del proyecto de investigación, se observo que la administración de proyectos de software es muy amplio, por lo que con respecto al objetivo general plantado, puede decirse que se trato de abarcar muchos aspectos que requerirán de una investigación mas profunda y que puede ser la pauta para futuras investigaciones.

Las administración de proyectos de software, será una de las actividades que más constantemente se realizara, que con el desarrollo de nuevas tecnologías, podrá ayudarse a tener un mayor control de cada proyecto realizado, y esta será una las tareas más demandantes en el mundo de la información como se le a dado en llamar a la era de la computación, ya que proporciona elementos de control de toda índole, ya que la información que hoy genera es el recurso mas protegido por todas las organizaciones.



## BIBLIOGRAFIA

- Ackoff Russell L.  
Planificación de la empresa del futuro  
Limusa. México. 1998
- Ackoff Russell L.  
El paradigma de Ackoff  
Limusa Wiley. México. 2002
- Ackoff Russell L.  
El arte de resolver problemas.  
Limusa. México. 2003
- Ardines bujano Aimee Lilian,  
"Planeacion y aseguramiento de la calidad en los proyectos de desarrollo de software"  
Tesis de Matemáticas Aplicadas Y Computación. 1994
- Braude Eric J.  
Ingeniería de software  
Una perspectiva orientada a objetos  
Alfa-Omega. México. 2003
- Bruegge Bernd, Dutoit Allen H.  
Ingeniería de software orientada a objetos  
Prentice Hall. México. 2002
- Clenand Davd I, King William R.  
Manual para la administración de proyectos  
CECSA, Mexico 2003.
- Chamoun Yamal,  
Administración profesional de proyectos, La guía,  
IAN Ediciones, México, 2002



Chiavenato Adalberto  
Introducción a la teoría general de la administración.  
Mc Graw-Hill, Colombia.

Churchman C. West  
El enfoque de sistemas  
Diana, Mexico, 1989

Domingo Ajenjo Alberto  
Dirección y Gestión de Proyectos  
Un enfoque práctico  
Alfaomega Ra-Ma, Mexico, 2000

Hernández Jiménez Ricardo  
Administración de la función informática  
FACTOR AFI  
Trillas, Mexico, 2003

Kendall y Kendall.  
Análisis y Diseño de Sistemas  
Prentice-Hall, México, 1994

Gido Jack y Clements James P.  
Administración exitosa de proyectos  
Internacional Thomson Editores, Mexico, 1999

Lawrence Pfleeger Shari  
Ingeniería de software, Teoría y Práctica  
Prentice Hall 2002

Pressman Roger S.  
Ingeniería del software "Un enfoque práctico"  
McGraw Hill, México, 1998



---

Robbins Stephen P. Coulter Mary  
Administración  
Prentice Hall. Mexico. 2000. 6ta Ed.

Ralph M.Stair. Reynolds George W.  
Principios de sistemas de información.  
Thomson editores. México. 2000

Sommerville Ian  
Ingeniería de software  
Adison Wesley 2002

Miklos Tomas. Tello Ma. Elena  
Planeación interactiva.  
Limusa. México. 2002

Miklos Tomas (Coordinador)  
Las decisiones políticas  
De la planeación a la acción.  
Instituto Federal Electoral. México. 2001

Reyes Ponce Agustín  
Administración Moderna  
Limusa. México. 2001

Romero Betancourt.  
Principios Fundamentales de Administración de Empresas  
Limusa. México, 2000.

Van Der Heijden KessEscenarios  
El arte de prevenir el futuro.  
Panorama. México. 2000



**Direcciones electrónicas consultadas:**

[www.alpha-hardin.com](http://www.alpha-hardin.com)

Roberto Toledo. Administre mejor sus proyectos, 12 pasos básicos

[www.ieee.org/portal/index.jsp](http://www.ieee.org/portal/index.jsp)

American Institute of Electrical Engineers.(IEEE).

[www.iso.org/iso/en/ISOOnline.frontpage](http://www.iso.org/iso/en/ISOOnline.frontpage)

Organización internacional de estándares Internacionales (ISO).

[www.milestone.com.mx/CursoAdmProysBreve.htm](http://www.milestone.com.mx/CursoAdmProysBreve.htm)

milestone consulting

[www.software.net.mx/inicio/default.htm](http://www.software.net.mx/inicio/default.htm)

Sitio oficial de la industria mexicana del software.

[www.sei.cmu.edu/cmm](http://www.sei.cmu.edu/cmm)

Carnegie Mellon Software Engineering Institute

Modelo de capacidad de madures.

[www.monografias.com](http://www.monografias.com)

Historia de la computación.