

## **6. GUIA Y ESTRATEGIAS PARA DESARROLLAR E IMPLANTAR UN SISTEMA CON CALIDAD.**

A través de esta guía la cual hace referencia al proceso de desarrollo de sistemas ideal, se ha decidido incluir las mejores técnicas (prototipo, modelación, JAD entre otras) y herramientas como CASE que aseguran no sólo el desarrollo de una aplicación de la manera más rápida, sino la participación del usuario además de un control en todo el proceso, logrando así un sistema con calidad.

Son muchas tecnologías las que se van a incorporar, varias ya conocidas y dominadas por algunos y otras como CASE que requerirán de un modelo o enfoque para implantarse como se vio en el capítulo 4.

Es importante considerar que para que esta guía tenga el éxito que se espera, las tecnologías incorporadas deberán asimilarse lo más pronto posible, siguiendo para ello algún proceso y contando con el apoyo de la alta administración; también se buscará lograr el entendimiento de esta guía y lo más importante es que las personas deberán estar convencidas de sus beneficios.

Para adoptarla se recomienda:

- Proporcionar un entrenamiento.
- Empezar a practicar con ella en proyectos pilotos.
- Mejorar la eficiencia con su uso.
- Ya una vez dominada será ejecutada sin ningún esfuerzo logrando excelentes resultados.

Acompañando a esta guía estarán por lo tanto, las posibles causas, que no deseamos que ocurran y sus cursos de acción para su posterior solución, esto será parte de la implementación del modelo de prevención indicado en el punto 5.7 de este proyecto.

Puede considerarse que suceda alguna causa y su curso de acción no esté contemplado, para lo cual deberá analizar cuáles son las causas que ocasionan determinado síntoma, para posteriormente determinar el curso de acción para su solución e implementarlo, actualizando esta guía para fomentar así, la mejora en futuros desarrollos.

Deberá considerar que todo cambio implicará una resistencia para lo cual tendrá que emplear los conceptos de administración de cambio, para lograr disminuir la resistencia cuando se haya conseguido aumentar el grado de aceptación.

Además sus miembros del equipo de desarrollo deberán de tomar en cuenta la experiencia de los:

- Analistas del área de soporte técnico o de telecomunicaciones según sea el caso, los cuales evaluarán el impacto que tendrá el futuro volumen de transacciones en relación al tiempo de respuesta de las terminales y al rendimiento que pueda tener la red, además podrán recomendar procesos para poder balancear la red.

- Analistas de operación, que ayudarán a definir los requerimientos tanto de memoria como de hardware, evaluarán el impacto de el nuevo volumen de transacciones en relación a los procesos que ya se ejecutan en ciertos períodos de tiempo, pudiendo además sugerir estructuras de los archivos de datos y formas de acceso para poder lograr una mayor eficiencia desde el punto de vista operacional, estimando el uso quizás de cintas y de discos indicando las características de desempeño máximo y normal, recomendando además procesos de recuperación y respaldo.

- Auditores, quienes dada su experiencia y conocimiento en :

- + Funciones del negocio.
- + Metodologías de desarrollo de sistemas.
- + Estándares para documentación.
- + Planeación de sistemas.
- + Administración de proyectos.
- + Etc.

Deberán sugerir técnicas de control o guías para cada fase compararlas con las que se han estado utilizando recomendando controles adicionales para fortalecer el medio ambiente de desarrollo. Siendo estos controles preventivos o detectivos y ya no correctivos. Su labor no será de revisar algo que ya esta hecho, ya que se realizarán las cosas bien a la primera sino que asesore al personal de desarrollo de sistemas para mejorar cada vez el proceso. ¿Por qué tener gente extra revisando lo que otros deben hacer bien?, esto resultaría muy costoso.

## **a) Planeación de Sistemas:**

### **Actividades:**

- Definición de objetivos.

Deberán de definirse los objetivos a nivel departamental, asimismo en este proceso deberán de participar los gerentes de los diferentes departamentos así como del área de sistemas, no olvide que los objetivos deberán ser específicos cuidando que sean realizables (no imposibles), concretos (claros), armonizados (deseables) y cuantificables (medibles) y lo cuales deberán estar relacionados de algún modo con los de la empresa.

- Deberán definirse los proyectos que apoyen los objetivos, considerando los proyectos generales del departamento y las solicitudes de los usuarios.

Es importante mencionar que estas solicitudes deberán de entregarse previo a la junta de planeación de sistemas y las cuales deberán de estar completas con respecto a la información requerida. Para ello esta solicitud elaborada por el usuario y revisada junto con él por un elemento del área de sistemas deberá estar apoyada por un anteproyecto que indique el objetivo, beneficios, tecnología propuesta, alcances, así como las actividades relevantes por subsistema que forman parte de él.

- Asignar prioridades, estas deberán ser asignadas por los gerentes o jefes del área(s)(clientes) involucrados con los sistemas. Utilice alguna técnica para darles prioridad.

- Asignación de recursos y fechas a los proyectos aceptados, esto deberá de registrarse haciendo uso de la herramienta CASE que cuenta con la opción de administración de proyectos. Esa fecha a su vez deberá ser considerada posteriormente en el plan detallado del proyecto para definir a su vez fechas y recursos de las actividades de las fases del desarrollo del nuevo sistema.

El apoyo de una herramienta CASE será muy importante ya que a través de ella se podrá establecer una estructura de trabajo (actividades), asignar tareas, tiempos y estimar costos utilizando el modelo de COCOMO. Estas herramientas además nos permiten manejar métodos como PERT-CPM y Gantt para controlar el avance del proyecto. Gracias a esta herramienta podremos:

- + Desarrollar y modificar con facilidad los diagramas de barras (Gantt) y de flujo (PERT-CPM), siempre resaltando este último el camino crítico.

- + Traducir los plazos del proyecto a fechas del calendario real, es decir tomando en cuenta los días festivos y vacaciones.

- + Con los diagramas de flujo que podremos construir lograremos:

- \* Apreciar con claridad todas las actividades, los eventos y las estimaciones de tiempo.

- \* Determinar fácilmente el tiempo en que puede realizarse el proyecto.

- \* Coordinar todas las actividades.
- \* Realizar análisis del tipo ¿qué puede salirnos mal? determinando así las consecuencias de una demora de cualquier actividad.

+ Y con los diagramas de Gantt podremos:

- \* Seguir y controlar la marcha del proyecto y suministrar una retroalimentación importante al equipo de desarrollo que interviene en el proyecto.
- \* Reflejar el presupuesto económico o las necesidades de personal o de equipo, nos ayudaría a evitar la sobre explotación de recursos sobre todo si van a ser compartidos en varios proyectos.
- \* También estos diagramas nos van a ser de mucha ayuda para realizar el seguimiento y control de los gastos reales o de la utilización real de los recursos.
- \* También es posible llevar a cabo análisis ¿Qué ocurriría si?, dibujándose automáticamente al introducir los datos la gráfica y así realizar los análisis para tomar la mejor decisión.

+ Con esta herramienta se podrá recuperar los plazos de un proyecto que se ha retrasado. Ya que si se observa algún margen de deslizamiento en un camino determinado se pueden trasladar los recursos sobrantes a las actividades del camino crítico mostradas por PERT, con el fin de agilizarlas.

- A final de cuentas si usted utiliza esta herramienta permitirá que el proyecto acabe dentro del plazo, del presupuesto y dentro de las especificaciones de calidad.

#### **Productos terminados:**

- Objetivos del departamento.
- Solicitudes de servicio que cumple con los requerimientos del departamento de sistemas (ver Anexo C).
- Anteproyectos de sistemas.(Información detallada en relación al sistema como objetivos, beneficios, tecnología propuesta, actividades relevantes por subsistemas y alcances).
- Asignación de fechas y recursos para realizar el diagnóstico para cada proyecto.
- Plan de sistemas.

#### **Involucrados:**

- Usuarios.
- Líder del proyecto.

- Responsable de sistemas.
- Auditor.

**Responsables:**

- Responsable de sistemas.

**Técnicas y herramientas:**

- Software administrador de proyectos (por ejemplo: Project Management).
- CASE soportando la parte de administración de proyectos.
- Técnica para dar prioridad a los proyectos. (Considerando las prioridades proporcionadas por el usuario).
- PERT-CPM o diagrama de flechas, GANTT.
- Documentación.
- Diagrama marcial, diagrama de afinidad, diagrama de árbol, diagrama de proceso de decisiones programadas, diagrama de relación, diagrama de pareto, hoja de verificación.

**Causas que ocasionan una deficiente planeación:**

- No se lleva a cabo la petición de un servicio a través de una solicitud formal.
- Usuario desmotivado por no cumplirle antes.
- Rechazo de solicitudes después de retenerla por varios días, ya que no cumple con las especificaciones requeridas por el departamento de sistemas. Información incompleta.
- Después de retener la solicitud de servicio por un tiempo éste no se lleva a cabo por que no es de alta prioridad, indicándole al usuario que no se realizará su proyecto.
- Por carecer de una planeación o por su mala práctica se desperdician recursos y muchas veces los proyectos que se planean no contribuyen a los objetivos del negocio.
- No se cuenta o no se asigna al personal idóneo con respecto al área a la que pertenece el proyecto a desarrollar.

**Cursos de acción a seguir:**

- Definir como política del departamento de sistemas que para dar servicio a un requerimiento o una necesidad del usuario, deberá realizarse una petición formal a través de una solicitud por escrito y firmada, presentándola además antes o en la reunión que se realiza para llevar a cabo la planeación de sistemas.

- Si la propuesta del usuario contribuye a los objetivos del negocio y apoya las estrategias de la empresa su proyecto será tomado en cuenta según la prioridad que el mismo usuario le dé atendándolo en base a los recursos con los que se cuente y a fechas asignadas.
- Si la información de la solicitud está incompleta o no es clara, planear una entrevista con el usuario para aclarar con él las dudas, de tal que la solicitud pueda ser completada.
- La recepción de solicitudes deberá realizarse en un período previo a la planeación estratégica de sistemas, de no ser así, las solicitudes que se reciban posteriormente podrán permutarse si el área en cuestión tiene asignado otro proyecto ya planeado y donde el tiempo y los recursos destinados a ese proyecto sean semejantes al nuevo.
- Deberá de manejarse una técnica para dar prioridad a los proyectos considerando ante todo los objetivos estratégicos de la empresa. Consultar y pedir opiniones antes de tomar una decisión.
- Dar una retroalimentación al usuario independientemente de cuál sea el resultado de la evaluación de su proyecto, (Aceptado alta prioridad, baja prioridad, o rechazado). Indicar las razones por su rechazo. Esto deberá de a través de un documento formal anexado a la solicitud de servicio, este documento deberá ir firmado por la persona que atendió la solicitud.
- Se deberán considerar los avances tecnológicos para mejorar este proceso de planeación, y registrarse todos aquellos proyectos en un administrador de proyectos con el fin de darles un seguimiento adecuado. Se registrará el nombre del proyecto y el responsable del mismo. Si no cuenta con la herramienta justifíquela haciendo una evaluación y preséntela en la siguiente etapa.

## **b) Diagnóstico:**

Deberá de analizarse la necesidad que ha sido presentada por el usuario, llevando a cabo un análisis preliminar del sistema actual, con el fin de determinar cuál es la problemática u área de oportunidad que se presenta. Se deberá de investigar el funcionamiento de todas las partes del sistema actual conociendo :entradas, procesos, salidas, tecnología que utiliza, áreas y gente involucrada. El tiempo que se le dedique a esta fase será determinado por el líder del proyecto, este dependerá mucho de la magnitud del proyecto y de la cantidad de recursos que éste disponga, lo que es importante recalcar es que esta fase siendo la base del desarrollo del sistema requerirá de más tiempo junto con el análisis que una fase de diseño, construcción y pruebas. Recuerde que si no se entiende el problema no será posible encontrar la solución.

### **Actividades:**

- El líder del proyecto seleccionará a las personas idóneas para llevar a cabo el diagnóstico. Estas personas deberán de ser analistas con amplia experiencia práctica en el campo de los negocios y con las siguientes habilidades:
  - + Guiar y ayudar a descubrir los problemas.
  - + Que sepa trabajar en equipo.
  - + Saber escuchar.
  - + Expresar ideas claras y en forma breve.
  - + Aceptar puntos de vista de otros.
  - + Vendedor de ideas.
  - + Ser observador y detectar lo que está oculto y que no ha salido a flote.
  - + Formular las preguntas adecuadas para obtener información relevante.
  - + Inspirar confianza en la gente y animarla a que digan lo que realmente piensan.
  - + Manejar técnicas y herramientas para obtener información relevante.
  - + Concretizar y sintetizar describiendo con objetividad los problemas.

Siendo todo esto indispensable si se desea que esta fase sea exitosa dando el diagnóstico correcto, reconociendo así los problemas y sus causas.

- El proceso a seguir para reconocer cuáles son los problemas y sus causas, será:
  - + Realizar un plan para llevar a cabo la recopilación de información.
  - + Seleccionar la(s) técnicas para efectuar la recopilación.

**Cuestionarios:** los cuales representan entrevistas autoaplicadas. A través de ellos es posible cuantificar las respuestas, utilizándolo sobre todo cuando la muestra de personas es grande.

**Entrevistas:** A través de la cual podremos descubrir la forma en que la organización funciona, es considerada como una fuente rica de información.

Observación: Por medio de esta podremos observar el comportamiento a medida que este ocurre, obteniendo la realidad.

- + Diseñar el cuestionario, las preguntas a realizar en la entrevista, así como definir que observar y a quienes según sea el caso.
- + Llevar a cabo la recopilación de información.
- + Una vez recopilada la información construya un modelo (de procesos y de datos) en una sesión de lluvia de ideas ( *brainstorming* ), auxiliado por una herramienta CASE que soporte esta actividad, utilizando técnicas de análisis estructurado, logrando así representar la situación real para poderla analizar y determinar tanto los problemas como sus causas. La herramienta CASE le será de mucha utilidad para construir un modelo de alto nivel de la empresa, así como el modelo conceptual del sistema utilizando el diagramador de procesos y de flujo de datos, ayudando al analista(s) a determinar las redundancias que existen en el proceso modelado.
- + Auxílese además de estadísticas representadas por gráficas, basándose en las herramientas estadísticas y administrativas de calidad (Anexos).
- + Analice y sintetice la información generando un diagnóstico parcial.
- + Presente al usuario los resultados mostrándole de ser posible los modelos de procesos creados.
- + Si el cliente acepta los resultados, este podrá a su vez retroalimentar considerando esta información, en un proceso de consolidación de información. Quizás se requiera una modificación a los diagramas ya elaborados esto no será problema si estos puedan ser corregidos inmediatamente al utilizar la herramienta CASE.
- + Consolide la información generando así un diagnóstico general indicando :
  - \* Los objetivos de la empresa.
  - \* La estructura actual de la organización , sus productos y servicios, áreas afectadas sus clientes, las políticas establecidas, información sobre sus métodos y procedimientos, la descripción de puestos y toda la información relacionada con el sistema actual (procesos, información tanto de entrada como de salida).
- + Lista de problemas y sus causas así como el objetivo del proyecto a realizar como una solución posible el cual deberá de ser verificado por el usuario, no olvide que este objetivo deberá ser realista, claro y medible.

#### **Productos terminados:**

- Datos generales de la empresa
- Objetivos de la empresa.
- Estructura organizacional. Descripción de puestos.
- Lista de usuarios y áreas afectadas.



- Políticas establecidas.
- Manuales de métodos y procedimientos.
- Información relacionada con el sistema actual como :
  - + Software y hardware en el que se desarrolla el sistema.
  - + *Escenario de seguridad que rodea al sistema.*
  - + Información que requiere y genera el sistema.
  - + Procesos que son cubiertos por el sistema.
- Modelo del negocio.
- Modelación de procesos y de datos del sistema actual.
- Estadísticas haciendo uso de las herramientas de calidad.
- Lista de problemas y sus causas.
- Objetivo del proyecto.
- *Plan de recopilación de información.*
- Cuestionarios, resultados de las entrevistas y de la observación.

**Involucrados:**

- Usuario.
- Analista.
- Auditor.
- Líder del proyecto.
- Areas de soporte al sistema.
- Proveedores/Clientes relacionados con el sistema.

**Responsables**

- Líder del proyecto.
- Analista.

**Técnicas y herramientas:**

- Entrevistas.
- Cuestionarios.
- Observaciones.
- Causa-efecto.
- Análisis estructurado.
- Documentación.
- CASE (Middle-CASE, Lower-CASE).

- Brainstorming.
- Diagrama de seis palabras.
- Pareto.
- Estratificación.
- Diagrama de dispersión.
- Gráfica de control.
- Diagrama de relaciones.
- Diagrama de árbol.
- Hojas de chequeo.
- Diagrama de afinidad.
- Diagrama matricial.
- Documentación.

**Causas que ocasionan un diagnóstico deficiente:**

- No se involucra al usuario.
- No hay un compromiso por parte del usuario, piensa que pierde su tiempo.
- El usuario tiene dificultad para comunicarse.
- El analista no conoce o no utiliza las técnicas y herramientas.
- El analista no entiende el problema.
- No se realiza un estudio del sistema actual.
- Inconsistencia entre la problemática especificada por el usuario y la determinada por el analista de sistemas.
- Objetivos no claros, cuantificables o controlables.
- No existe la documentación del sistema anterior.

**Cursos de acción a seguir:**

- *El usuario es una pieza clave para entender el problema y conocer todos los procesos e información que éstos requieren y producen. Deberá siempre involucrarse si se desean obtener excelentes resultados.*

- El usuario a veces no se compromete por temor a ser desplazado o porque se resiste al los cambios que puedan ocurrir. Hay que convencer al usuario de que se le ayudará a realizar mejor su trabajo para que así éste pueda desempeñar otras funciones.
- Se requerirá de habilidad por parte del analista para obtener la información que se desea cuando el usuario tiene problemas para comunicarse. Se le recomienda diseñar un cuestionario afín a la problemática que lleve al analista a localizar, y enfocar en forma correcta y concreta el problema existente.
- Capacitación en el uso de las metodologías, técnicas y herramientas para que conozca ¿qué son? ¿Cuándo y cómo utilizarlas ? además de conocer los beneficios que con ellas se pueden lograr.
- Para que el analista entienda mejor la problemática que se presenta en una determinada área, deberá estudiar antes aquellos procesos que puedan estar relacionados con el área a investigar (indicada en la solicitud de servicio), y además deberá apoyarse con bibliografía para entender aquellos tecnicismos que no se conocen o que en un momento dado no hayan sido bien explicados.
- El analista deberá conocer cual es la técnica (entrevistas, cuestionarios, observación) más adecuada para recopilar la mayor cantidad de información.
- En un caso dado que la problemática entre ambas partes (usuario y analista) tenga un enfoque muy diferente, llevar a cabo sesiones extras para resolver esas diferencias y aclarar dudas, buscando que el enfoque de los problemas sea uniforme y realista.
- Muchas veces los objetivos no son claros porque el analista no ha comprendido ni la problemática, ni los procesos involucrados en ellos. Si es necesario deben aclararse nuevamente las dudas teniendo breves sesiones con los usuarios de las áreas involucradas.
- Si no existe la documentación se llevará más tiempo el realizar un análisis del sistema actual, sin embargo si éste existiera no hay que olvidar que se debe registrar en una herramienta CASE, o si ya existe, podrá reutilizarla para elaborar la nueva modelación de datos y procesos para el nuevo sistema. En dado caso que sólo se cuente con el diseño físico de la base de datos, puede aplicarse la ingeniería en reversa con CASE para generar así el modelo entidad-relación deseado.

### **c) Definición de requerimientos y desarrollo de la propuesta.**

Definir y documentar los requerimientos que se pretenden obtener por medio del sistema. Tal que se determine:

- Quién es el que necesita la información.
- Qué información necesita, cuándo y dónde.

Logrando en base a esos requerimientos desarrollar la propuesta del proyecto a presentarse para su posible aceptación.

#### **Actividades a seguir:**

- El analista junto con el usuario deberá de definir y documentar los requerimientos de:

Funcionalidad.  
Desempeño.  
Documentación.  
Seguridad.  
Calidad.  
Datos.  
Medio ambiente.  
Organizacionales.

(a) Tomando como base la información recopilada en la etapa anterior así como los diagramas de flujo de datos y entidad-relación, y quizás información de entrevistas y/o cuestionarios extras, podremos determinar cuales son los requerimientos, y (b) construir un prototipo el cual esté basado en esos requerimientos encontrados inicialmente. (c) Una vez construido el prototipo auxiliado por una herramienta CASE, el usuario podrá interactuar con él de tal manera que se determinen nuevos requerimientos o cambios en los diseños y formatos de entrada y salida. (d) Posteriormente el analista realizará las modificaciones pertinentes repitiendo el proceso (c) y (d) hasta que el usuario quede satisfecho.

Como construir el prototipo:

Utilizando herramientas CASE:

- + Podrá definir las entidades o archivos a utilizarse en las pantallas y los reportes, esto se hará si no han sido definidos en el modelo entidad-relación del sistema actual. (Considere los puntos de vista del analista de operación).
- + Crear las pantallas y reportes definiendo títulos, encabezados y campos previamente definidos, si desea que todos los campos estén incluidos solo basta hacer referencia al archivo o entidad generándose estos automáticamente en el reporte o en la pantalla.
- + Crear los menús que harán referencia a esas pantallas y reportes.

- + *Crear el prototipo final* indicando la secuencia de menús, pantallas y reportes. soportado el prototipeo por CASE es posible construir y modificar el prototipo rápidamente ayudando a que el proyecto no se retrase de acuerdo a lo planeado, logrando una calidad e integridad de la información.

En dado caso que no tenga una herramienta CASE que soporte el prototipeo, puede hacer uso de un lenguaje 4GL o algún software de aplicación (DBASE, FoxBase, Access) utilizando sus generadores de pantallas y reportes. Será indispensable utilizar estas herramientas cuando necesitemos realizar cálculos especiales o manejar cierta lógica que muchas veces no es posible con las herramientas CASE. Para construir su prototipo con este tipo de software deberá:

- + Construir sus archivos.
- + Construir sus menús.
- + Construir la pantallas para dar entrada a información.
- + Construir los reportes.
- + Construir otras funciones quizás procesos batch, llamadas a lenguajes de tercera generación, cálculos matemáticos etc.

Y seguir el mismo proceso de verificar con el usuario y modificar el prototipo hasta que quede listo. Si la herramienta CASE que usted posea cuenta con un lenguaje 4GL o si le permite interfaces con algún otro lenguaje esto le ayudará a construir un prototipo que satisfaga las necesidades de sus usuarios.

La calidad que se logre obtener de los requerimientos se obtendrá si éstos llegan a refinarse a través del prototipo.

- Elaboración de la propuesta. Considere al elaborar la propuesta la información y restricciones que le pueda proporcionar el analista de operación y de telecomunicaciones.

En base a los requerimientos previamente especificados se deberá definir:

*Objetivos particulares del proyecto. Deberán estar relacionados con los requerimientos definidos.*

*Alcances.*

*Limitaciones.*

*Diferentes alternativas para solucionar el problema.*

- + *Mantenimiento.*
- + *Desarrollo de un nuevo sistema.*
- + *Compra del sistema.*
- + *Continuar con el sistema como está.*

*Definir para cada alternativa:*

- + *Factibilidad económica: Realizar un análisis costo-beneficio, (utilizando técnicas de comparación de costos y beneficios así como aquellas relacionadas con la estimación de costos), estimando el costo de operación del sistema actual, el costo de operación del sistema propuesto e indicar aquellos beneficios tanto tangibles como intangibles*

*determinando de esta manera si los costos pueden compensarse por las ganancias a largo plazo.*

*+ Factibilidad técnica: Determinar si los recursos técnicos como el hardware, software y el soporte técnico que se pueda tener satisfacen los requerimientos definidos.  
De no ser así, determinar si la actual tecnología se necesita modernizar o si se requiere de una nueva, tomando en cuenta que esto pueda ser justificable financieramente.  
Realizar una evaluación de hardware y software según se requiera seleccionando aquel que satisfaga más las necesidades.*

*+ Factibilidad operativa. Ver si se cuenta con el personal capacitado para soportar el desarrollo de sistemas, de no ser así determinar si es viable capacitarlo.*

*+ Factibilidad de recursos humanos. Independientemente si se va a requerir de un nuevo sistema o de una modificación a uno ya existente hay que recordar que esto provocará en la mayoría de los casos una resistencia al cambio por parte de las personas afectadas directamente por el sistema. Es importante considerar si estas personas pueden obstaculizar el desarrollo o implementación del sistema, determinando el problema que representan, cambiando la resistencia que estas personas puedan tener basándose en la presentación de las ventajas que se pueden obtener por medio del sistema. A pesar de esto, puede persistir la resistencia y por lo tanto debe convencerse, o de ser necesario, presionar para lograr los fines que se requieran.*

Realizar el plan general del proyecto indicando:

- + Actividades a realizar.
- + Responsables.
- + Fechas de inicio y terminación de cada actividad, considerando los compromisos de inicio del sistema por requerimientos externos.
- + Metodologías, técnicas, herramientas y estándares propuestos.
- + Plan de calidad.
- + Plan de documentación.
- + Plan de pruebas.
- + Plan de entrenamiento.
- + Plan de seguridad(contingencias y recuperación).

Haciendo uso de la herramienta CASE, deberán de registrarse las fases y las actividades para la realización de los diferentes proyectos propuestos, así como los responsables indicando la fecha de inicio y terminación, las horas a dedicarle y el costo total por cada una de las actividades. Cabe mencionar que ésta información no podrá ser registrada si el proyecto no ha sido dado de alta en el sistema.

Registrando esta información el administrador del proyecto podrá consultar el avance del mismo conociendo el estado del proyecto e identificando las desviaciones del plan para llevar a cabo las acciones adecuadas con el fin de asegurar los resultados deseados.

- Preparar presentación de la propuesta.
- Presentación de la propuesta (Vender el proyecto).
- Rechazo o aceptación del proyecto por parte del usuario o de la gerencia a la que éste pertenece y del responsable de sistemas.

**Productos terminados:**

- Documentación conteniendo los requerimientos.
- Propuesta de desarrollo.
- Plan del proyecto.
- Documento firmado de aceptación del proyecto.

**Involucrados:**

- Gerencia a la que pertenece el usuario.
- Usuario(s).
- Analista de sistemas, de operación y de telecomunicaciones.
- Líder de proyectos.
- Responsable de sistemas.
- Auditor.

**Responsables:**

- Líder del proyecto.
- Analista.

**Técnicas y herramientas a utilizar:**

- Un software administrador de proyectos. (por ejemplo: Project Management).
- GANTT.
- PERT-CPM o diagrama de flechas, para apoyar la planeación y control de proyectos si no cuenta con la herramienta CASE.
- CASE con soporte a la administración de proyectos (Categoría Upper-CASE).
- Diagrama de pareto.
- Métodos de estimación de costos (COCOMO).

- Técnicas de comparación de costos y beneficios:
  - + Análisis del punto de equilibrio: Utilizarse cuando se necesite justificación por costo y no por los beneficios.
  - + Análisis del flujo de efectivo: hacer uso de éste cuando el proyecto es muy costoso.
  - + Valor presente: Utilizarse cuando el período de inversión es largo o si el costo del financiamiento es elevado.
  - + Retorno de inversión: Se recomienda su uso si los beneficios tangibles mejoran.
  
- Utilizar técnicas para recopilación de información (requerimientos).
  - + Cuestionarios.
  - + Entrevistas.
  
- Técnicas para identificar los requerimientos.
  - + Prototipeo.
  - + JAD.
  
- Documentación.

**Causas que ocasionan una definición de requerimientos y una propuesta de desarrollo deficiente:**

- Que no se realice el análisis de requerimientos.
- No se conocen las técnicas, herramientas o metodologías para realizar esta actividad.
- Equipo inadecuado para el uso de las herramientas, no soporta el software.
- Demora en la aceptación de proyectos.
- Requerimientos incompletos.
- Requerimientos no viables.
- Persisten los problemas con el usuario para obtener la información.
- Objetivos no claros, cuantificables o controlables.

**Cursos de acción:**

- Para obtener la información acerca de los requerimientos no se debe olvidar utilizar las técnicas como la entrevista o cuestionarios.



Si se ha decidido realizar una entrevista se le recomienda que lleven a cabo los siguientes pasos:

- a) Lectura de antecedentes.
- b) Establecer el objetivo de la entrevista.
- c) Seleccionar a los entrevistados.
- d) Preparación del entrevistado: Es importante hacer sentir al entrevistado que la información que proporcione será de gran utilidad para poder lograr la calidad en el proyecto a desarrollar. Además de esto, deberá de obtenerse una cita y el entrevistador le deberá indicar cuáles serán los diferentes puntos a tratar.
- e) Estructurar las preguntas

Además de tomar notas para documentar la entrevista, puede apoyarse de una grabadora para registrar completamente todo lo que mencione el entrevistado, además con el uso de la grabadora puede realizar más preguntas y obtener así más información. Salvo que el entrevistado no este de acuerdo en el uso de la grabadora hay que aceptarlo.

Si se ha decidido aplicar un cuestionario deben considerarse los siguientes puntos que le indican la conveniencia en su uso:

- a) Las personas a quienes necesita interrogar se encuentran localizadas en diferentes áreas.
- b) Existe una gran cantidad de usuarios, es indispensable conocer sus necesidades.
- c) Es necesario llevar a cabo un estudio exploratorio.

- Vender la idea acerca de los beneficios que se logran con el uso de las metodologías, técnicas o herramientas.

- A través de una actualización continua en tecnología de información, evalúelos y presente los resultados a la alta dirección para su aprobación.

- Se recomienda construir un prototipo con el cual el usuario pueda trabajar para determinar si los requerimientos obtenidos son los que se desean y además detallarlos. Esto podrá ser soportado por una herramienta CASE( Categoría Lower-CASE) con el fin de mejorar la comunicación con los usuarios cuando sobre todo los requerimientos no son claros, detallando así los requerimientos en un tiempo más corto, como resultado de la interacción con el prototipo. No es recomendable para sistemas de tipo batch o sistemas que incluyen procesos de cálculo complejos.

- Deberá adoptarse como una política del departamento, que todo proyecto deba ser presentado con los lineamientos anteriormente mencionados (Objetivos, alcances, limitaciones, costo-beneficio, etc.) Es muy importante que el costo y beneficio esté presente, ya que de no ser así, no se podrá tomar una decisión para continuar con el desarrollo del proyecto. Debe considerarse los siguientes puntos para realizar un análisis costo-beneficio.

- a) Identificar los costos y los beneficios tanto tangibles como intangibles.
- b) Llevar a cabo estimaciones para los costos y beneficios tangibles.
- c) Y por último utilizar una de las metodologías anteriormente mencionadas para llevar a cabo la comparación de los costos y los beneficios.

No olvide que a lo largo de todo el proyecto haciendo uso de la herramienta CASE que soporta la administración de proyectos los miembros del equipo de desarrollo registrarán todas aquellas actividades que se vayan terminando, tal que el líder del proyecto pueda verificar el avance del proyecto, realizando los ajustes necesarios con el fin de lograr un sistema dentro del tiempo y del presupuesto previsto. Los cambios tendrán que ser informados al equipo de desarrollo de manera clara, oportuna y comprensiva; estos podrán ser dados a conocer fácilmente teniendo acceso a la información que maneja la herramienta CASE la cual se encuentra almacenada y actualizada en su repositorio. Asimismo el usuario podrá realizar consultas teniendo acceso a la información almacenada en el repositorio de CASE, conociendo así el grado de avance de su proyecto.

#### **d) Análisis:**

Actividades:

d.1) Modelación de datos.

El modelo de datos que se definirá será la base del desarrollo del sistema .

Debe quedar claro que el éxito de esta actividad dependerá no sólo de que el usuario esté involucrado en cada paso en el proceso de la modelación, sino de que entienda perfectamente el modelo y lo revise, respondiendo el analista rápidamente a las sugerencias asistido por una herramienta CASE en sesiones JAD y además deberá de buscarse que el modelo esté normalizado.

Para construir el modelo o diagrama entidad-relación se tomará como referencia los preceptos de Chen. Con este diagrama se mostrarán todas las entidades y sus datos indicando además la manera como estas se relacionan.

Al construir este modelo estaremos separando el diseño conceptual de datos del diseño físico para posteriormente una vez elegida la base de datos donde vamos a implementar el modelo este podrá convertirse en un diseño físico.

Haciendo uso del editor que posee la herramienta CASE usted podrá hacer referencia al modelo de datos previamente creado en la fase de diagnóstico, si desea modificarlo, o podrá crear uno nuevo tal que las entidades sean representadas por rectángulos, las relaciones por rombos y mediante las líneas pueda conectar los conjuntos de entidades al conjunto de relaciones, mostrando además la cardinalidad.

Su construcción será muy fácil ya que a través de una paleta que maneja la herramienta se muestran los diferentes símbolos tal que usted podrá ir seleccionando el símbolo que desee, así hasta terminar de construir su diagrama.

Además para definir los atributos que compongan a cada entidad deberá indicar su nombre, su formato y si es o no llave. Gracias a la herramienta CASE podremos darnos cuenta de aquellas asociaciones no definidas basándose en las llaves definidas como foreign, reportando además las redundancias. Todo el modelo quedará almacenado en un repositorio para su uso posterior como será su normalización, su conversión a un esquema deseado o así mismo podrá ser accesado para hacer referencia a él cuando se realice modelo conceptual del sistema.

He aquí los pasos para construir el modelo utilizando la herramienta CASE la cual soporta la técnica de análisis estructurado, involucre al analista de operación y de telecomunicaciones:

- Se deberán de tomar en cuenta los requerimientos de información definidos en la etapa anterior, considere además el modelo de datos ya construido del sistema actual, almacenado en el repositorio.
- Identifique cuales serán las entidades a formar parte del nuevo esquema conceptual, podrá modificar el modelo ya creado o crear uno nuevo, considere que para la realización del

prototipo ya fueron definidas las entidades o archivos basta hacer referencia a ellas al estar formando el modelo entidad-relación.

- Establecer la relación entre las entidades con los rombos e indique la cardinalidad.
- Normalice el modelo. La herramienta CASE podrá normalizar el modelo automáticamente hasta la tercera forma normal, asegurando así la corrección, exactitud, eficiencia y estabilidad. Logrando además no sólo que el conjunto de entidades sean precisas y no redundantes sino que las relaciones complejas se dividan en simples.
- Cheque el modelo con los usuarios.  
Dado que este tipo de modelos es fácil de entender ya que se acerca más a la realidad es necesario que este sea verificado con el usuario, el analista de operación y de telecomunicaciones, para determinar que cumple con sus requerimientos.
- Resolver las discrepancias.
- Modificar el modelo.
- Repetir las actividades a partir de la normalización hasta que el usuario esté de acuerdo. Se recomienda realizar esto en una sesión JAD.
- Convierta el diagrama entidad-relación o esquema conceptual a un esquema DBMS si lo desea o al manejador de archivos a utilizar, pudiéndolo transferir a donde resida el software DBMS o manejador de archivos.

Que quede claro que las estructuras de la base de datos podrán cambiar en un futuro pero el objetivo de esta modelación será el de minimizar esos cambios. Las estructuras en el negocio deben ser estables, mientras que los procesos tienden a cambiar constantemente.

#### d.2) Modelación de procesos.

Es importante determinar cuáles serían los procesos que utilizarían todas las entidades definidas en el modelo de datos, generado en la actividad anterior.

##### Actividades:

La herramienta CASE a través de su módulo modelador de diagramas de flujo de datos (DFD) y de procesos podrá soportar todas las siguientes actividades a realizar tal que la herramienta a través de su poderoso editor gráfico nos permitirá crear estos diagramas utilizando los símbolos básicos que representan los procesos, flujo de datos, almacenamiento de datos, y entidades externas, siendo los procesos explotados a múltiples niveles. Lo más importante que se obtiene al utilizar esta herramienta es que por la fácil manipulación de símbolos y su fácil uso se logra una máxima productividad, siendo la comprobación de errores de sintaxis y tipo, así como la verificación de la consistencia y el balanceo una de las funciones más importantes de la herramienta asegurando así la integridad y exactitud de los diagramas de flujo de datos.

Todos los diagramas serán almacenados en un repositorio pudiendo ser reutilizados para futuros proyectos ya sea en su forma parcial o total.

- Tome como referencia antes de construir el nuevo diagrama de flujo de datos, información sobre los procesos involucrados así como otras soluciones puestas en práctica por otras empresas.
- Realice la descomposición del proceso en subprocesos hasta llegar al detalle generando así los diagramas de procesos.
- Asignar nombres a los procesos de preferencia la composición del nombre será (verbo+ nombre del objeto) .
- Definir la relación entre los procesos y realizar un diagrama de dependencia de procesos.
- Definir una matriz en la cual se establezca la relación entre procesos y entidades previamente ya definidas y validadas y verificadas por el usuario.
- Realizar el diagrama de contexto.
- Realizar el diagrama de flujo de datos tomando en cuenta la matriz previamente definida. Generando diagramas hasta el nivel más bajo que se requiera, de esta manera estará utilizando la técnica top-down la cual consiste en describir más detalladamente las burbujas o procesos del DFD cero. Puede hacer referencia al DFD creado previamente en la etapa de diagnóstico.  
Una de las mayores ventajas al estar utilizando herramientas CASE para la producción de estos diagramas es que en las sesiones JAD y llevando a cabo una lluvia de ideas pueden llegar a esbozarse varias opciones de solución utilizando el diagrama de flujo de datos y estando presente él o los usuarios puede llegarse a la mejor solución, no olvide considerar las políticas de la organización. Recuerde que ya no se tratara de automatizar lo que se estaba realizando si no de ver cual opción nos ayudará a mejorar y a la vez a reducir costos, sin olvidar el cumplimiento de los requerimientos antes definidos.
- No olvide numerar cada uno de los procesos que están definidos en el diagrama de flujo de datos, indicando el flujo de datos entre ellos.
- Redibujar el diagrama de flujo de datos para llevar a cabo correcciones como resultado de las revisiones de éste con el usuario, con analistas de operación y telecomunicaciones.
- Realizar el diccionario de datos. (Describir los flujos de datos así como los eventos tanto del modelo interno como del externo). Esto será generado automáticamente por la herramienta CASE.
- Verificar el balanceo y la consistencia , las herramientas CASE tienen una opción para llevarse a cabo esta actividad automáticamente, esta opción puede activarse o desactivarse según lo requiera, evitando:
  - + Evitar que los procesos sean infinitos (que no tengan una salida).
  - + Evitar procesos de generación espontánea (que no tengan entradas).
  - + Los datos de entrada y salida en los procesos hijos deben estar también en sus procesos padre.
  - + La información que se defina en el diagrama de flujo de datos debe encontrarse en el diccionario de datos.
- Realizar las miniespecificaciones. Esta son aplicables a aquellos procesos que no se descomponen en más subniveles de detalle, estas describen las actividades que se realizan en

los diagramas de flujo de datos para convertir las entradas y salidas, utilizar para ello el español estructurado. Algunas herramientas no tienen la opción de manejar el español, así que tendrá que trabajar en inglés.

- Realice los diagramas de transición de estados. Para representar la conducta del sistema, cuando este sea un sistema de tiempo real.

d.3) Definir la información relevante tanto de entrada como de salida para cada proceso. Existen templates que maneja la herramienta CASE para definir la documentación de sus pantallas reportes, formas y archivos. De hecho al generar sus pantallas, reportes y archivos toda la documentación relacionada a estos estará almacenada en un repositorio pudiéndola imprimir en el momento que se desee, estando siempre actualizada.

La información recomendable a definir será:

- Nombre, objetivo, tipo de forma (manual o por sistema), formato, fuente, frecuencia de uso, para formas.
- Nombre, objetivo, tipo de pantalla (actualización y consulta), frecuencia de uso, para pantallas.
- Nombre, objetivo, formato, fuente, frecuencia con que se genera, para reportes.
- Nombre, Objetivo, tipo de archivo, fuente, frecuencia de uso, para archivos

d.4) Actualización del plan.

Quizás se requiera de actualizar o cambiar el plan general, tal que se requiera de más recursos, como resultado del análisis.

d.5) Realizar la reevaluación de costos.

Si se requiere de más recursos para la realización de este proyecto, realice nuevamente estimaciones de costos, no olvide utilizar el método de COCOMO.

d.6) Reevaluar la justificación.

Dado los nuevos costos que no se habían tomado en cuenta será necesario actualizar el documento de análisis costo-beneficio empleando nuevamente técnicas de comparación de costos.

d.7) Obtener la aceptación del usuario o del cliente y del responsable de sistemas. Deberá analizarse el nuevo análisis costo-beneficio, si lo hay. El usuario determinará si se continúa o no con el proyecto, determinado si es rentable o no el proyecto y si los costos y beneficios no han cambiado. Deberá ser formalizado a través de un documento.

#### **Productos terminados:**

Toda la documentación relacionada con estos productos será generada automáticamente a través de la herramienta CASE, ya que ésta solo hará acceso al repositorio de información donde todos estos elementos se encuentran. Con excepción del análisis costo-beneficio y del documento formal de aceptación.

- Esquema conceptual (Modelo de datos de alto nivel).
- Modelo de datos de alto nivel normalizado.
- Esquema conceptual (Modelo de datos en un específico DBMS o en un manejador de archivos).
- Esquema interno (para el mismo DBMS o manejador de archivos).
- Diagrama de procesos (DP).
- Diagrama de dependencia de procesos (DDP).
- Matriz entidad-proceso (MEP).
- Defina el diagrama de contexto (Modelo externo)
- Diagrama de flujo de datos (DFD Modelo interno).
- Miniespecificaciones por cada proceso (ME).
- Diagrama de transición de estados del sistema (DTE).
- Diccionario de datos (DD).
- Toda la documentación relacionada con : archivos, pantallas, reportes y formas.
- Plan general actualizado.
- Análisis costo-beneficio actualizado.
- Documento final que indique la aceptación de usuario y del responsable de sistemas.

**Involucrados:**

- Líder del proyecto.
- Analista de sistemas, operación y de telecomunicaciones..
- Usuarios.
- Auditor.
- Operación.

**Responsables:**

- Líder del proyecto.
- Analista.

**Técnicas y herramientas a utilizar:**

- Entrevistas. Si no se realizan sesiones JAD.
- Análisis estructurado.
- Técnica de modelación.
- Técnica de normalización.
- Técnica Top-down.
- Software administrador de proyectos (por ejemplo Project Management).
- JAD.
- Case (Categoría Middle-CASE) soportando el análisis el cual siempre es estructurado..
- CASE soportando la administración de proyectos.
- PERT-CPM o diagrama de flechas.
- GANTT.
- Diagrama matricial, diagrama de relación, causa-efecto.
- Técnicas de comparación de costos.
- Estimación de costos (COCOMO).
- Documentación.

**Causas que ocasionan un análisis deficiente:**

- No se entienden los requerimientos especificados en la etapa anterior.
- No se cuenta con la información suficiente para identificar entidades y llevar a cabo la relación entre ellas y con los procesos.
- No se conocen, no se usan o no se saben utilizar las técnicas y herramientas. Muchas veces no se utilizan por que la herramienta no es clara y simple.
- Esquema conceptual está demasiado normalizado.
- Mala definición de índices.

**Cursos de acción:**

- Llevar a cabo entrevistas si la definición de requerimientos fué realizada por otras personas, para aclarar dudas.
- Cursos de capacitación apoyados por el responsable de sistemas.

- Determinar si la herramienta es la adecuada por medio de una evaluación, de no ser así justificar la compra de otra. Es indispensable para evitar la no realización de la documentación por lo tedioso que pueda implicar su ejecución a mano. Además muchas veces los cambios recomendados no se llevan a cabo si no se cuenta con una herramienta. Por ejemplo si trabaja con los diagramas de flujo podrá realizar:

- + Cambios continuamente.
- + Diagramas de flujo de datos a diferentes niveles.
- + Verificar la consistencia y el balanceo.

- Hay que determinar si es conveniente la normalización ya que algunas veces se disminuye el rendimiento en el sistema.

- Hay que verificar en el momento de realizar el diseño físico que los índices sean adecuados.



### **e) Diseño.**

A través del diseño se podrá construir el modelo conceptual computacional del sistema necesario para su programación. Conociendo el problema y los requerimientos, se logra definir la solución computacional utilizando para ello técnicas y herramientas gráficas.

#### **Actividades:**

- Realizar el diseño lógico del sistema, considere la información e inclusive las posibles restricciones dadas a conocer por el analista de operación y de telecomunicaciones.

Por medio de la herramienta CASE podrán generarse automáticamente los diagramas estructurados accedando al diagrama de flujo de datos (DFD) encontrado en el repositorio o bien si la herramienta no tiene esta poderosa función será a través del editor de la herramienta como tendrá que construir los diagramas estructurados.

Para ello deberá:

- + Definir la estructura general del sistema.
- + Para cada módulo utilice la técnica top-down para dividirlo en más módulos pequeños especificando el módulo al detalle que desee, tome como referencia el DFD ya creado. Será necesario indicar el flujo de información tanto de datos como de control en las llamadas de un módulo a otro.  
Las herramientas CASE utilizan el rectángulo para representar el módulo, flechas con un círculo relleno o no relleno indicando el flujo de control o de información respectivamente añadiendo un identificador para darle un significado al flujo.  
Debido a que la lógica de cada módulo no se especifica debe detallarse cada módulo indicando su nombre, la lista de parámetros de entrada y salida, el valor de retorno del módulo y su función.

CASE verificará la exactitud y eficacia del diseño estructurado indicando aquellas inconsistencias y eliminándolas.

Será indispensable para asegurar la calidad:

- + Utilizar técnicas de diseño estructurado soportado por la herramienta CASE.
- + Una vez construidos los diagramas estructurados determinar si existe un bajo grado de acoplamiento (parámetros que existen entre los módulos de todo el diagrama) asegurando así su reusabilidad, y el alto grado de cohesión (integración de las instrucciones) para asegurar su facilidad de cambios.
- + Al realizar el diseño la división realizada en módulos nos asegura un cumplimiento de los factores de calidad: flexibilidad, facilidad de prueba, portabilidad.

- Realice los casos prueba.

El diseñador deberá definir tanto los datos de entrada a cada uno de los programas a probarse así como las funciones del programa a ejecutarse con esos datos y la propuesta del resultado esperado con el fin de que sean tomados en cuenta al hora de realizar las pruebas para verificar y validar tanto los programas que conforman al sistema, como el propio sistema una vez ya integrados estos.

La herramienta CASE puede agilizar este proceso generándole los datos prueba tomando como base la estructura del programa, pero es hasta la siguiente etapa cuando puede utilizar la herramienta. Sin embargo las funciones a realizar y los resultados tendrían de todos modos que definirse.

- Crear opciones técnicas.

Las opciones tanto de hardware como de software deben haberse contemplado en el estudio de factibilidad. Si la decisión no ha sido tomada, deberá de tomarse ahora. Para ello se debe hacer un esquema de las posibles opciones a nivel muy general y discutirlos con los usuarios, analistas de operación y telecomunicaciones definiendo entre todos la mejor estrategia a seguir.

Asimismo deberá de revisar la lista de requerimientos con objeto de comprobar si las opciones posibles cumplen dichos requerimientos. Por último deberá seleccionar la mejor opción en base a un análisis costo-beneficio.

Es indispensable que recuerde que es conveniente utilizar la misma herramienta CASE que ha estado manejando durante el transcurso de este desarrollo de sistema u otra que se pueda integrar a las anteriores para poder generar el código de los programas automáticamente y codificar ciertos detalles realizando interfases con otros lenguajes. Considere en sus soluciones el software CASE generador de código y otros lenguajes de tercera o cuarta generación si se requieren.

- Realizar el diseño físico.

- Actualización del plan.

Quizás se requiera de actualizar o cambiar el plan general, tal que se requiera de más recursos, como resultado del análisis.

- Realizar la reevaluación de costos.

Si se requiere de más recursos para la realización de este proyecto realice nuevamente estimaciones de costos, no olvide utilizar el método de COCOMO.

- Reevaluar la justificación.

Dado los nuevos costos que no se habían tomado en cuenta será necesario actualizar el documento de análisis costo-beneficio empleando nuevamente técnicas de comparación de costos.

- Obtener la aceptación del usuario o del cliente y del responsable de sistemas. Deberá analizarse el nuevo análisis costo-beneficio. El usuario determinará si se continúa o no con el proyecto, determinado si es rentable o no el proyecto y si los objetivos, costos y beneficios no han cambiado. Deberá ser formalizado a través de un documento.

**Productos terminados:**

Esta documentación será generada automáticamente por la herramienta CASE, a través de su módulo de reportes o accedando cada módulo de la herramienta para imprimir así cada producto generado por el módulo. Con excepción del análisis costo-beneficio y el documento de aceptación.

- Diseño lógico de programas (diagramas estructurados) estructuras de información.
- Casos prueba.
- Diseño físico.
- Plan general actualizado.
- Análisis costo-beneficio actualizado.
- Documento final que indique la aceptación de usuario y del responsable de sistemas.

**Involucrados:**

- Líder del proyecto.
- Analista de sistemas, operación y telecomunicaciones.
- Diseñador.
- Usuarios.
- Auditor.

**Responsables:**

- Líder del proyecto.
- Analista.

**Técnicas y herramientas a utilizar:**

- Top-down.
- Diseño estructurado.
- Herramienta CASE (Categoría Middle-CASE, Lower-CASE) soportando el diseño estructurado para la construcción de los diagramas estructurados.
- CASE soportando la administración de proyectos.
- PERT-CPM o diagrama de flechas.
- GANTT.

- Técnicas de comparación de costos.
- Documentación.

**Causas que ocasionan un diseño deficiente:**

- El análisis que se tiene es deficiente o no se cuenta con documentación.
- Herramientas no existen o no se utilizan por que no se tiene conocimiento sobre ellas o no son eficientes.
- Las técnicas que se pueden utilizar en esta etapa son desconocidas.
- Cambio de técnica con relación a la utilizada en el análisis.
- Diseño deficiente, de programas es incompleto y no cumple con las especificaciones establecidas.
- El diseño de pruebas no se realiza.

**Cursos de acción:**

- Puede concertarse una cita para aclarar dudas. Esto no deberá de suceder si usted desde el principio del desarrollo se utiliza una herramienta como CASE y además si las actividades tuvieron la calidad esperada.
- Hacer uso de una herramienta CASE, ya que al hacer los diagramas estructurados tendrán que realizarse:
  - + Cambios continuamente.
  - + Realizar los diagramas estructurados descomponiendo en módulos a diferentes niveles.
  - + Verificar la consistencia.

de no ser así, resultará más costoso hacer correcciones en una etapa más avanzada como lo serían las pruebas.

- La capacitación continua evitará que las técnicas y herramientas sean desconocidas.
- Deberá utilizarse la misma técnica empleada en el análisis para que éste sea congruente con el diseño.
- Utilizar una herramienta CASE para validar y verificar el diseño estructurado.
- Se deberán diseñar los casos prueba asegurando que existe una correspondencia entre la aplicación y los requerimientos. Diseñar casos prueba que cubran todas las alternativas posibles; este diseño de prueba reduce costos. CASE sólo podrá generar datos prueba si tiene ya la estructura del programa.

### **f) Construcción y pruebas por unidad:**

En esta etapa de desarrollo de sistemas, es necesario llevar a cabo la traducción de las especificaciones del diseño a código fuente, y efectuar las pruebas a cada uno de los programas generados y que conforman el sistema.

#### **Construcción.**

Tomando en consideración la herramienta ya seleccionada y con la cual se va a construir el sistema no olvide utilizar si éstos son lenguajes de tercera o cuarta generación técnicas de programación que sean consistentes, con las técnicas manejadas en el análisis y diseño como:

**Estructurada:** Permite realizar programas más fáciles de escribir, verificar, leer y mantener utilizando un número limitado de estructuras de control como lo son: secuencial, selectiva y repetitiva. Además incorpora tanto recursos abstractos y diseño top-down .

**Modular:** A través de este tipo de programación es posible ser más productivos logrando dividir en programa en módulos, realizando cada uno de ellos una actividad o tarea específica, y donde cada uno de ellos es analizado, codificado y probado por separado.

**Top-down:** El programa a codificar se realiza descomponiendo el problema a resolver en niveles, tal que se considere cada estructura desde dos puntos de vista, ¿lo que hace?, y ¿cómo lo hace?.

Al programar en lenguajes de tercera o cuarta generación tendrá que:

- + Tomar como base el diseño estructurado creando programas para cada uno de los módulos que ahí se proponen.
- + Al codificar no olvide manejar estándares y de tener un buen estilo de programación.
- + Sus programas deberán ser :
  - \* Legibles :Fáciles de leer y comprender.
  - \* Portables: Fáciles de instalar en cualquier plataforma.
  - \* Modificables: Para realizar los cambios necesarios.
  - \* Eficientes: Aprovechar al máximo los recursos de la computadora minimizando la memoria utilizada y el tiempo de proceso o ejecución.
  - \* Modulares: Dividirlos en bloques donde cada uno de ellos realice un aparte del conjunto del trabajo.
  - \* Estructurados: Cumpliendo reglas de la programación estructurada para facilitar su verificación y depuración.

Si ha tomado la decisión de utilizar el generador de código de la herramienta CASE, el código generado automáticamente y consistente con el diseño estará ya libre de errores, lo que no sucederá si programa utilizando lenguajes procedurales (3GL) o no procedurales (4GL).

Una vez generado el código, puede editarlo utilizando cualquier editor de líneas, en este caso el de la herramienta, por si se desean agregar más procesos.

Al terminar la edición podrá compilar el código y después se podrá generar la aplicación tal que el código compilado se ligue a las librerías adecuadas para crear una aplicación ejecutable libre de errores. Asimismo esta aplicación podrá ser instalada en el medio ambiente que desee.

Debe considerar independientemente de si utiliza una herramienta CASE o un lenguaje procedural, la reutilización de aquellas subrutinas o librerías que han sido almacenadas para hacer referencia a ellas, estandarizadas para que sean fáciles de aplicar y validadas para su fácil integración con el resto del programa. Si reutilizamos esta librerías o código lograremos beneficios como:

- Velocidad.
- Disminución de costos.
- Calidad( Ya que esas librerías están no sólo probadas sino quizás hasta mejoradas).
- Reducción de mantenimiento.
- Documentación debido a que los componentes reusables están bien documentados.

Por último no olvide realizar el plan detallado de pruebas por unidad registrándolo en un administrador de proyectos, considerando las fechas para prueba del plan general, asimismo las actividades que vaya realizando deberá ir las registrando, llevando acabo comparaciones de lo real contra lo planeado realizando ajustes para que se logre terminar el proyecto dentro del tiempo y presupuesto acordado.

### **Pruebas por unidad.**

Las pruebas son realmente importantes ya que de esta manera valoraremos la calidad, determinando el apego de los programas a las especificaciones ya definidas, estas deberán realizarse si sus programas han sido construidos por un lenguaje procedural ya que al ser generados por una herramienta CASE podrán probarse y ya no como pruebas por unidad sino como una prueba del sistema ya que como se cuenta con toda la aplicación solo tendríamos que comprobar su calidad realizando las pruebas con casos previamente diseñados.

Si sus programas están en un lenguaje procedural, considerar lo siguiente:

- Tomar en cuenta los casos prueba que se generaron en la etapa de diseño o utilice herramientas CASE que generen los datos pruebas basándose en sus programas o en el dominio de los datos de entrada al programa, si no las usa capture sus datos prueba y los resultados generando dos archivos respectivamente.

- Realizar primeramente las pruebas para investigar las propiedades estructurales del código fuente.

Utilice una herramienta CASE del tipo analizador estático para detectar errores estructurales como variables no inicializadas, errores entre parámetros formales y reales e incumplimiento de estándares. Esta prueba deberá de repetirse hasta que no existan errores .

- Realizar las siguientes pruebas de caja negra haciendo referencia a la base de datos de prueba y de resultados ya generadas para poder investigar el comportamiento del código fuente sobre los datos prueba.

- + Funcionales: Conocidos los datos de entrada y los resultados esperados verificar los resultados obtenidos. Siendo estas pruebas apoyadas por las herramientas CASE catalogadas como verificadores de prueba o herramientas de prueba dinámica.

- + **Desempeño:** Determinar el tiempo de ejecución, la eficiencia del programa, tiempo de respuesta y uso de dispositivos. Utilice herramientas de prueba CASE llamadas analizadores de tiempo y de optimización.
- + **Tensión:** Tratar de romper intencionalmente la unidad o módulo.

Por último en lo que respecta a pruebas por unidad realice las pruebas de caja blanca como lo es la de:

- + **Estructura:** En base a los casos prueba decidir qué rutas dentro del módulo se verificarán. Utilice las herramientas CASE verificadores de prueba.

Al terminar de realizar estas pruebas estaremos verificando el cumplimiento del factor funcionalidad en cada programa.

Considere además de las herramientas como CASE, utilizar:

- + **Comparadores de archivo** para verificar los resultados esperados contra los resultados obtenidos, tal que se reporten las diferencias.
- + **Rastreadores de programa** los cuales proporcionan información del comportamiento de todos los datos en todo el programa o módulo.
- + **Code auditor:** verifica las estructuras y el código.
- + **Test auditor:** Determina el número de saltos o transferencias realizados y no realizados durante la prueba.
- + **Generador de diagrama de flujo:** Genera un diagrama de flujo del código actual , a través del cual se muestra dónde los datos son utilizados para su posterior análisis

Recordar que se tratará de encontrar errores no de ocultarlos, hay que tener la mente destructiva si se requiere el éxito de las pruebas.

Por último deberá documentar sus programas siguiendo estándares y realizar el plan detallado de pruebas del sistema sin olvidar de utilizar su herramienta para la administración del proyecto, registre aquellas actividades que ya se cumplieron, realizando una comparación contra el plan general para llevar a cabo acciones ajustes para que se logre terminar el presupuesto dentro del tiempo y presupuesto acordado.

**Productos terminados:**

- Listado de programas documentados.
- Base de datos prueba.
- Resultados de las pruebas y de resultados esperados
- Programas validados, depurados y verificados.

- Reporte que muestre la diferencia entre los resultados de las pruebas para cada programa.
- Plan de pruebas detallado.
- Plan actualizado, por los ajustes sin afectar el tiempo y el presupuesto definido para el proyecto.

**Involucrados:**

- Líder del proyecto.
- Analista.
- Diseñador.
- Programador.
- Auditor.

**Responsables:**

- Líder del proyecto.
- Programador.

**Técnicas y herramientas:**

- CASE generadores de datos prueba.
- Editores de línea.
- Librerías o biblioteca de rutinas.
- Generadores de código (Categoría Lower-CASE).
- Herramientas CASE (Analizadores de tiempo, optimizadores y verificadores).
- Lenguajes de cuarta generación.
- Técnicas de programación (Top-down, estructurada, modular).
- Lenguajes de programación procedurales .
- Analizadores estáticos de programas.
- Paquetes de rastreo de programas.
- Pruebas de caja negra.
- Pruebas de caja blanca.
- Herramientas Code-Auditor, Test-Auditor.



- Generador de diagramas de flujo del programa.
- CASE soportando la administración de proyectos.
- PERT-CPM o diagrama de flechas, GANTT.
- Software para generar archivo de datos prueba y resultados.
- Documentación.

**Causas que ocasionan una construcción y pruebas por unidad deficiente:**

- Mala interpretación de las especificaciones del análisis y diseño.
- Estilo y lenguaje de programación complejo, dada la falta de estándares de programación, lo cual implica que no se logre un código de calidad uniforme entre todos los programadores.
- Se pierde mucho tiempo en traducir el diseño a código ya que no es semejante.
- Código no portable.
- No existe documentación interna en los programas.
- No se realizan las pruebas por unidad.
- Al realizar las pruebas no se cuenta con los casos ni con los datos prueba.
- No se cuenta con los manuales del sistema operativo ni del lenguaje de programación o de las herramientas seleccionadas como CASE.
- No se utilizan las técnicas y herramientas.
- Las especificaciones son inadecuadas.
- No existe validación a los datos de entrada.
- No se utilizan técnicas de programación.
- Fallas a leer o grabar información, acceso incorrecto a los elementos de un arreglo, variables no definidas, ciclos infinitos, abuso de variables, condiciones invertidas e incompletas, anidamientos inadecuados, incorrecto paso de parámetros.

**Cursos de acción:**

- Existe mala interpretación de las especificaciones porque las dudas que se tienen no son aclaradas. Por lo tanto los programadores, deberán aclarar sus dudas con el analista o diseñador, si no se llevará acabo una mala traducción del diseño al codificar.
- Si sólo se dispone de un lenguaje de programación no hay nada que hacer. Sin embargo para que su lenguaje no sea complejo debe considerarse al elegirlo:

- + Area de aplicación.
- + Complejidad algorítmica.
- + Entorno en el que se ejecutará el lenguaje.
- + Consideraciones de rendimiento.
- + Complejidad de las estructuras de datos.
- + Disponibilidad de un compilador.

En lo que respecta al buen estilo de programación, éste inclusive puede superar las deficiencias de un lenguaje de programación primitivo; si su estilo es pobre de nada servirá el excelente lenguaje que tenga.

Su estilo de programación será bueno si el código es fácil de comprender, si es sencillo y elegante.

Las acciones a evitar para poder tener un buen estilo de programación serán:

- + No debe ser demasiado complicado.
- + Si maneja instrucciones condicionales evitar then o then if o else nulos.
- + No realizar muchos anidamientos ni en ciclos ni en instrucciones condicionales.
- + No suboptimizar.
- + No emplear identificadores para propósitos múltiples.
- + Utilizar librerías ya existentes.

Además haga uso de los estándares definidos por el departamento o que están definidos por la metodología seleccionada:

- + Instrucciones.
- + Indentación si se requiere.
- + Reglas asociadas con los módulos a programar.
- + Nombres de datos.
- + Nombres de módulos.
- + Nombres de archivos.
- + Nombres de registros.
- + Etc.

Son indispensables los estándares para que el estilo de codificación sea similar es por eso que se deben definir y documentarse para uso de los programadores. Si se utilizan herramientas automatizadas será fácil verificar el uso de los estándares, además al no utilizarlos será difícil entender los programas desarrollados por otras personas.

- Al elegir un lenguaje seleccionar el que más se adecúe a su diseño o a la técnica de diseño elegido, ya que si no, dedicará mucho tiempo en la traducción.

- Si la aplicación a desarrollar tendrá que ser transportada a otro procesador, compilador o trabajar con una nueva versión de sistema operativo o inclusive ser utilizado por otros paquetes, será la portabilidad un requisito indispensable.
- El que no exista documentación interna causará problemas al realizar un mantenimiento al programa, es por eso que al terminar el programa debe incluirse la siguiente información por programa y por cada módulo que lo conforma:

*Por programa:*

- + Nombre del programa.
- + Tarea que realiza.
- + Nombre del programador.
- + Fecha en que se creó el programa.
- + Número de versión o actualización.
- + Fecha de la última actualización.
- + Lista de módulos del programa indicando quién llama a quién.
- + Conexiones con otros programas.
- + Entradas y salidas.

*Para cada módulo definir:*

- + Su función.
- + Descripción de variables.
- + Autor del módulo.
- + Fecha de la última modificación.

Además por cada programa no se debe olvidar utilizar:

*Comentarios descriptivos en el programa para:*

- + Describir los bloques del código.
- + Dejar líneas en blanco y sangrías para que el código sea entendible y legible.

- Las pruebas de cada módulo y programa deberán llevarse al terminar de codificar para lo cual deberán de tomarse en cuenta los casos prueba previamente diseñados así como los datos de prueba también ya creados, de modo que se eliminen todos los errores si la prueba resulta ser exitosa.
- Se deben tener en el área de trabajo los manuales necesarios para trabajar y sobre todo tenerlos actualizados y en buen estado.
- Muchas veces los programadores trabajan a mano, hay que tomar en consideración los editores tan poderosos que existen actualmente, sobre todo los que acompañan al lenguaje de programación. El hecho de utilizar los editores es por que a veces manejan comandos que reducen el tiempo que requiere el programador para editar su programa, las herramientas que rastrean el programa o los analizadores estáticos le ayudarán rápidamente a encontrar sus errores.

- Para que las especificaciones no sean inadecuadas deberá utilizar lo propuesto en la fase de diseño.
- Muchas veces no se utilizan técnicas de programación como la estructurada, modular o la orientada a objetos, porque no se conocen o porque no se sabe cómo aplicarlas, las técnicas de programación deberán ir acorde a lo definido en el diseño. En un dado caso que la técnica no se domine, deberá de capacitarse.
- Deberán construirse los procesos que permitan validar cualquier información que se introduzca al sistema, más aún si el lenguaje no realiza por el tipo de datos que maneje, una validación automática.
- El abuso de variables, ciclos infinitos, exceso de anidamientos etc., es causa de la falta de experiencia en el lenguaje, y en programación puede evitarse esta problemática si utiliza lenguajes de cuarta generación o herramientas CASE.

### **g) Pruebas del sistema:**

En esta fase se realizarán acabo tanto la prueba del sistema, como la prueba de aceptación por parte del usuario, asegurando así que el sistema cumpla con los requisitos y rendimiento solicitado por el usuario.

#### **Actividades:**

- Generar la base de datos prueba utilizando una herramienta CASE o tomando como referencia los casos prueba diseñados anteriormente tal que capture los datos y cree el archivo tanto de datos como de resultados.
- Realice la prueba de integración .  
Para integrar todos los módulos o programas, verificar la interfase entre ellos y asegurar que cada módulo así como los que se vayan integrando cumplan con los requerimientos y especificaciones del diseño. Utilice para ello los siguientes enfoques:
  - + Top-down: Se empieza probando el módulo raíz y posteriormente todos los demás módulos subordinados se van llamando y probando hasta que todos los módulos sean incorporados.
  - + Bottom-up: Cada uno de los módulos del nivel más bajo se prueba individualmente, después los siguientes módulos son llamados por los ya probados hasta que todos los módulos se han incluido en la prueba.
  - + Big-bang: Ya probados todos los módulos individualmente todos se encadenan y se realiza la prueba.
  - + Sandwich: Combina tanto el enfoque top-down como el bottom-up. Los niveles de arriba se combinan con los del centro y los de abajo también. Utilizando el enfoque to-down para el nivel superior y el bottom-up para el inferior.

Haga uso de herramientas de comparación, CASE verificador y Drivers que simulan la llamada a módulos e inclusive las ejecutan. Este tipo de herramientas agilizarán el proceso de esta etapa.

- Probar el sistema realizando:

- + Pruebas de recuperación: Hacer fallar al sistema verificando que la recuperación sea apropiada siendo ésta quizá en forma automática o mediante la intervención de una persona. Evaluar el tiempo de recuperación.
- + Pruebas de seguridad: Verificar si el sistema consta de mecanismos de protección, tratar de penetrar al sistema y determinar si el costo de penetración es mayor que el valor de la información obtenida mediante la penetración.

También será importante verificar la seguridad que el sistema proporcione, identificando así los riesgos de éste al cometer algunos errores en los datos de entrada, para ello será importante utilizar técnicas como:

*Análisis de árbol de fallos:* Mediante éste será posible determinar las consecuencias de una secuencia de fallos.

*Lógica de tiempo real:* Mediante un modelo sucesión se prueba la seguridad de los componentes del sistema.

*Redes de Petri:* Determinan los riesgos más p

- + Pruebas de resistencia: A través de esta prueba se enfrentará al sistema a situaciones anormales, tal que el sistema al ejecutarse demandará muchos recursos.
- + Pruebas de rendimiento: Asegurar el rendimiento deseado del sistema, rendimiento que sea probado desde la prueba por unidad.

- Realizar el plan detallado de la prueba de aceptación el cual deberá ser revisado y aceptado por el usuario.
- Generar la base de datos prueba para llevar a cabo la capacitación y la prueba de aceptación del sistema.
- Plan de capacitación detallado a algunos usuarios para prueba del sistema.
- Preparación del material de capacitación (hardware, software, manual del usuario y los documentos que firmará el usuario si es aceptado el sistema).
- Capacitación a sólo algunos de los usuarios siendo posibles entrenadores.
- Realizar la prueba de aceptación para determinar si el sistema satisface los requerimientos del usuario, validando la facilidad de uso del sistema .

Puede llevar a cabo:

- + Pruebas alpha: aquí el sistema es probado por un sólo usuario en el lugar de desarrollo. Ahí el líder del proyecto tomará notas de los problemas o errores en el uso del sistema.
- + Pruebas beta: el sistema es probado en diferentes lugares por diferentes usuario, estos registrarán los problemas y errores y se lo harán llegar al líder del proyecto.

- Realizar mínimas correcciones en caso de ser necesarias.

- Los miembros del equipo de desarrollo que intervengan en ésta etapa deberán registrar aquellas actividades que van terminando, el líder del proyecto realizará pequeños ajustes como resultado de las desviaciones al comparar lo real contra lo planeado. Esto no deberá afectar el tiempo y presupuesto del proyecto.

**Productos terminados:**

- Manual del usuario terminado.
- Manual técnico.
- Resultados de las pruebas.
- Base de datos prueba y de resultados esperados.
- Reporte que muestre las diferencias entre los resultados esperados y reales del sistema.
- Sistema depurado, validado y verificado.
- Documento formal que certifique la aceptación del sistema por parte del usuario.
- Plan general actualizado, con mínimos ajustes para que no se afecte el tiempo y presupuesto definido para el proyecto..
- Plan detallado de pruebas de aceptación.
- Plan detallado de capacitación.

**Involucrados:**

- Responsable de sistemas.
- Personas que realicen pruebas de integración y del sistema(audidores), en este tipo de pruebas se debe excluir a los desarrolladores del proyecto.
- Usuarios.
- Programadores.
- Auditores.
- Líder del proyecto.

**Responsables:**

- Líder del proyecto.
- Programadores.

**Técnicas y herramientas:**

- Generadores de datos prueba.
- Software administrador de proyectos.

- Herramientas CASE del tipo verificador y generadores de datos prueba.
- Pruebas alfa y beta.
- CASE soportando la administración de proyectos.
- PERT-CPM o diagrama de flechas y GANTT.
- Documentación.

**Causas que ocasionan deficiencia en las pruebas:**

- No existen casos pruebas o son inadecuados, si existen, muchas veces no se generan los datos prueba de acuerdo a los casos.
- No se detectan errores o existen problemas para detectarlos.
- No existe una planeación de las pruebas.
- No se realizan las pruebas.
- No se conocen las técnicas y herramientas para llevar a cabo las pruebas.
- Material de capacitación no existe.
- Capacitación deficiente.
- Manuales confusos e incompletos.
- Las personas que capacitan no tienen las cualidades necesarias para capacitar y convencer a los usuarios de los beneficios del sistema.

**Cursos de acción:**

- Es recomendable que el diseño de los casos pruebas se realice en la fase de diseño y que cubran todas las posibles alternativas dentro del sistema; los datos deben de generarse en base a estos casos.
- Si no se tienen casos prueba así como las herramientas para hacer las pruebas puede no detectarse ningún error.
- Deberá realizarse la planeación de las pruebas ya que si no, éstas no se realizarán por empalme de actividades o por que ya no hay tiempo para llevarlas a cabo. Debe indicar en este plan:
  - + Objetivo de las pruebas.
  - + Cómo las pruebas serán administradas y que criterios serán utilizados para juzgar cuando la prueba está completa.
  - + Deberán indicarse las herramientas y las técnicas a utilizar con esta prueba.
- Se deberán realizar las pruebas al sistema si se quiere evitar que los errores aparezcan durante su operación, logrando la insatisfacción del usuario.



- Vuelve el problema de la falta de investigación en nuevas tecnologías así como la capacitación en ellas.
- El material para capacitación puede ser un manual, material para clase, un video e inclusive un sistema con iconos o menús pop-up que explique el funcionamiento del sistema, es muy importante tomar en consideración el nivel educativo de las personas que capacitará.
- Deberá de preparar datos para el usuario para visualizar el comportamiento de éstos a través del sistema.
- Los manuales del usuario pueden empezarse a elaborar desde que el prototipo es aceptado. Si estos están incompletos pueden ocasionar que no se despejen las dudas o que sean confusos . Por lo tanto se recomienda que contengan por lo menos la siguiente información:

+ Introducción.

Funciones que ejecuta.  
Desarrolladores del sistema.  
Nombres de las personas que pueden atender sus dudas.  
Requerimientos de hardware y software.  
Limitaciones del sistema.

+ Principios y procedimientos.

¿Cuáles son las salidas? y ¿Cómo deben ser interpretadas?  
¿Cuáles son las entradas? y ¿Como se obtienen?  
¿Cómo se opera el sistema? y el tipo de menú e interfases que maneja.

Procesos generales.  
Formatear discos.  
Respaldar información.  
Entrada y salida del sistema.

+ Tutorial : mostrar paso a paso el funcionamiento del sistema con un ejemplo usar narraciones y mostrar las pantallas, así como los datos de entrada y salida.

+ Referencia.

Para cada función :

Describirla.  
Indicar al estructura de comandos.  
Indicar cuándo y como utilizarla.  
Mostrar que errores pueden ocurrir y qué hacer en caso de que ocurran.  
Ejemplos. '

Por error.

¿Cómo se identifican?  
¿Qué significan?  
¿Qué se puede hacer?

+ Como instalar el sistema antes de su primer uso.

- + Indicarle al usuario como crear sus propias aplicaciones (para generar la información deseada, hacer consultas a la base de datos mediante *Queries*).
- No debe ser muy técnico, ni muy teórico en la capacitación, debe hacerse la presentación lo mas atractivo que pueda ser utilizando un lenguaje accesible y detallado.

## **h) Implantación:**

En esta fase se terminarán de capacitar al resto de los usuarios, se instalará y probará el equipo, se convertirán o se construirán los archivos de datos verificando que el sistema quede instalado en la empresa. Esta etapa deberá estar apoyada por los vendedores, o alguien externo y el staff de desarrollo ya que es necesario que sea realizada por el departamento de operación liberando así de estas actividades al personal de desarrollo.

### **Actividades:**

- Capacitar al implementador del sistema. Este deberá conocer los aspectos generales del sistema, lugares donde será instalado, sus usuarios, los beneficios que con este sistema se logran así como el hardware y software que utilizará.
- Evaluar los conocimientos adquiridos del implementador.
- Verificar que el sistema sea de calidad y haya sido aceptado por el o los usuarios. Esta actividad deberá omitirse para el sistema desarrollado con la guía anterior ya que si no se estaría incurriendo en una inspección final innecesaria. Lo que si deberá de revisar será el documento que certifique la aceptación del sistema por parte del usuario.

Más sin embargo si su sistema fue comprado puede realizar una evaluación del sistema determinado su cuenta con los factores de confiabilidad, facilidad de uso, funcionalidad, realizabilidad, seguridad e interoperabilidad los cuales serán fácilmente visibles al ejecutar el sistema. Quien más que un usuario para evaluarlo asignando una puntuación de 0 a 5 a cada uno de estos factores, sumando los puntos y promediándolos determinando así si el sistema es:

CALIDAD	PUNTOS
<i>Excelente</i>	5
<i>Muy bueno</i>	$\geq 4$ y $< 5$
<i>Bueno</i>	$\geq 3$ y $< 4$
<i>Regular</i>	$\geq 2$ y $< 3$
<i>Pésimo</i>	$\geq 0$ y $< 2$

Si el factor de funcionalidad no llegara a cumplirse su sistema no deberá considerarse con calidad.

- Realizar el plan detallado de instalación, revisarlos con los usuarios y el responsable de sistemas de tal manera que se obtenga su autorización.
- Preparar el ambiente técnico para la instalación. Asegurar que el mobiliario y el equipo de cómputo está instalado y probado correctamente.
- Preparar si se requiere el material para capacitación de capturistas , dar aviso formal al período de captura.
- Capacitar a los capturistas, si se requiere.

- Conversión de datos o preparar los datos y capturarlos.
- Plan de capacitación al resto de los usuarios.
- Tener preparado el material de capacitación( equipo de cómputo, manual de usuario, material para exposición) así como los datos con los que se trabajará.
- Capacitar al resto de los usuarios.
- Instalación del sistema.
- Registrar aquellas actividades que fueron realizadas para posteriormente compararse contra el tiempo que se había planeado para ellas y realizar pequeños ajustes.

Para llevar a cabo esta instalación se podrá seguir cualquiera de las siguientes alternativas:

- Cortar totalmente el sistema antiguo y empezar con el nuevo. La instalación podrá efectuarse fuera de horarios de oficina.
- Operar un paralelo desfasado para dar confianza, fijando la fecha de inicio.

**Involucrados:**

- Responsable de sistemas.
- Líder del proyecto.
- Analistas.
- Diseñadores.
- Programadores.
- Usuarios.
- Consultor.
- Vendedores.
- Operación.

**Responsables:**

- Líder del proyecto.
- Operación.

**Técnicas y herramientas:**

- Utilerías para conversión de archivos.
- CASE soportando la administración de proyectos.
- Software para la administración de proyectos.

**Productos terminados:**

- *Plan de implantación detallado.*
- Plan de capacitación.
- Material de capacitación.
- Nuevas bases de datos o archivos.
- *Plan actualizado, por mínimos ajustes sin llegar a que afecte el tiempo y presupuesto total definido para el proyecto..*

**Causas que ocasionan la deficiencia en la instalación.**

- *El sistema no tiene calidad por consecuencia de un mal análisis, diseño y programación, por consecuencia el sistema no debe haber sido aceptado por el usuario.*
- Material de capacitación no existe .
- Capacitación deficiente.
- No se cuenta con el manual técnico actualizado.
- Manuales confusos e incompletos.
- *Las personas que capacitan a los usuarios no tienen las cualidades y habilidades necesarias para capacitar y convencer a los usuarios de los beneficios del sistema:*
- No se prepara el ambiente técnico para la instalación.
- No se conoce donde instalar el sistema.

**Cursos de acción:**

- *La persona que realizará la instalación del sistema deberá asegurarse de que el sistema tenga la calidad para lo cual deberá conocer totalmente el sistema. No deberá instalarse si este no ha sido aceptado por el usuario, para lo cual deberá verificarse la existencia de un documento formal que certifique su aceptación.*
- *Deberá preparar con tiempo el material para la capacitación si no se cuenta con el material de la fase anterior, este material puede ser un manual, material para clase, un video e inclusive un sistema con iconos o menús pop-up que explique el funcionamiento del sistema. Es muy importante tomar en consideración el nivel educativo de las personas que capacitará.*

- No se debe ser muy técnico, ni muy teórico en la capacitación, debe hacerse la presentación lo más atractiva que pueda utilizando un lenguaje accesible y detallado.
- Deberán prepararse datos para que el usuario pueda visualizar el comportamiento de éstos a través del sistema.
- A lo largo del desarrollo del sistema cada una de las personas del equipo de desarrollo deberá ir generando su documentación actualizándola por los cambios que puedan ocurrir, tal suerte que al final se tenga un manual completo y actualizado para futuros mantenimientos.
- Deberá de prepararse el ambiente técnico para instalar con anticipación el hardware y quizás una nueva versión del software de que será utilizado como plataforma, asimismo debe de considerarse el tiempo planeado de llegada del equipo y del software, el tiempo que se requerirá para su instalación para que pueda instalarse el nuevo sistema en la fecha convenida.
- Es indispensable conocer físicamente donde se llevará a cabo la instalación del sistema dentro de la empresa, para lo cual se tendrá que leer la información generada en el análisis.

### **i) Proporcionando garantía.**

Después de que el sistema ha pasado de la etapa de producción a la de operación ya no evaluaremos si el nuevo cumple con los requerimientos, ya de antemano se sabe que éste ya no tiene errores.

Sin embargo el departamento de sistemas establecerá un período de garantía de 2 a 6 meses teniendo opción a cualquiera de las siguientes alternativas:

- El líder del proyecto podrá permanecer en el lugar donde reside el sistema en operación.
- Hacer una llamada a cualquier persona que ha estado involucrada en el desarrollo del proyecto para atenderlo en forma inmediata.
- Llamar al departamento de sistemas para reportar el problema tal que este se registre y se haga llegar a la persona más indicada para atenderlo en el menor tiempo posible.

Usted podrá definir si se selecciona una de estas alternativas para todo el período de tiempo o si se realiza una combinación de todas en ese período.

Si surgen nuevos requerimientos o solicitudes de cambios, documéntelos quizás eso forme parte de un nuevo proyecto y no trate de ofrecer un mantenimiento al proyecto viejo.

#### **Involucrados:**

- Líder del proyecto.
- Usuario(s).
- Staff de desarrollo.

#### **Responsables:**

- Algún miembro del staff de desarrollo y el líder del proyecto.

Conociendo los objetivos de la calidad, los problemas que se presentan en el área de desarrollo de sistemas y contando ya con una guía para ese desarrollo se ha decidido que las siguientes estrategias son las más adecuadas para afrontar los problemas en el área de desarrollo con el fin de ir logrando la calidad en nuestros sistemas.

1. Organización: Establecer el concepto del proceso de calidad con la alta administración (directivo o responsable de sistemas) determinando cuál sería la organización necesaria del área de desarrollo de sistemas para apoyar este proceso.

Al utilizar una herramienta CASE en el desarrollo de sistemas sólo se requerirá de un verdadero líder del proyecto que realmente sea el administrador del equipo siendo el líder y motivador, quién planeará las actividades a llevarse a cabo, manteniendo un control con el fin de lograr un proyecto exitoso.

Se requerirán además analistas con la capacidad de diseñar interfases y de realizar modelaciones tanto de datos como de procesos, así como analistas-programadores con la capacidad de diseñar y construir programas sobre todo cuando estos sean desarrollados utilizando lenguajes 3GL o 4GL.

2. Formación: Crear en todo el personal una nueva cultura tanto de calidad como computacional o de informática ( en metodologías, técnicas y herramientas) basado el primero en programas de información al personal apoyado por la alta dirección y a través de una motivación hacia un deseo común. En cuanto a la cultura de informática esta se logra a través de un programa continuo de capacitación mejorando y actualizando los conocimientos y habilidades de las personas, previniendo riesgos e incrementando la calidad y productividad. Ese programa de capacitación será elaborado conforme a un diagnóstico de necesidades considerando la situación real y la situación ideal a la que se quiere llegar, llevando a cabo un análisis comparativo (relación de perfiles contra puestos). Será necesario también que el programa sea realizado considerando no sólo las necesidades del personal de sistemas sino sus deseos.

El nivel y escolaridad que se requiere que posean las personas involucradas en el desarrollo e implantación del sistema es de licenciatura o superior, en el área de sistemas computacionales requiriendo además de amplia experiencia y conocimiento del negocio.

Por ejemplo un Licenciado en Informática el cual puede evaluar, seleccionar e implantar un sistema computacional . Un Licenciado en Sistemas Computacionales es aquel capaz de diseñar, desarrollar e implantar los sistemas computacionales para administrar la información necesaria para la toma de decisiones , además de diseñar métodos y procedimientos que contribuyan a optimizar los recursos de la empresa. Y el Ingeniero en Sistemas Computacionales podrá desarrollar aquellos sistemas más complejos como sistemas operativos, manejadores de bases de datos tal que estos sean no sólo considerados como un software de aplicación sino como una nueva tecnología. El conocimiento de varios lenguajes de programación y el desarrollo de nuevas metodologías lo caracterizan.

3. Comunicación: Deberá existir una comunicación permanente entre los miembros del equipo de desarrollo de sistemas, para tener presente el avance que se ha logrado con la calidad en lo que respecta al proceso de desarrollo de sistemas e implantación.

4. Desarrollo en equipo: Todo proyecto que involucre la realización de un sistema de información tomando en consideración su magnitud, deberá ser realizado por un equipo de personas altamente capacitadas y de calidad.

5. Monitoreo: Dar seguimiento en lo que respecta a la ejecución del proyecto, detectando desviaciones conforme a lo planeado, analizando sus causas y corrigiendo las desviaciones con previa autorización.

6. El usuario del sistema: Deberá de ser la prioridad absoluta para el departamento de sistemas. La satisfacción se logrará a través de la calidad del sistema desarrollado.



7. Participación de los proveedores de información: Serán importantes contribuidores al desarrollo de nuevos sistemas con calidad. El nivel de calidad del sistema dependerá de la calidad de los requerimientos obtenidos. Si se desea desarrollar un sistema de información en menos tiempo y altamente confiable, se requerirá de su alta participación .

8. Proceso: La calidad deberá asegurarse a través del control del proceso del desarrollo de sistemas aplicando la técnica adecuada como la prevención. Una prevención que implicará hacer las cosas bien a la primera, además de anticiparse a los problemas conociéndolos al igual que sus causas, si llegara a suceder un problema no contemplado se deberán buscar sus causas eliminándolas lo más pronto posible desde su origen. No se tratará de lograr la calidad por inspecciones sino de fabricarla con el mínimo costo donde cada persona haga bien su trabajo y lo autocontrole.

9. Nunca parar: Se deberá buscar una mejora continua en los procesos, tiempo, gente y la tecnología.

- Proceso: Mejorar el proceso a través del uso de nuevas tecnologías de información.
- Tiempo: Reducción en tiempo de desarrollo y en costos.
- Hombre: Formación de grupos y equipos de trabajo.
- Tecnología: No copiar ideas de sistemas ya existentes: tratar de simplificar de transformarlos.

No se trata de que la mejora sea súbitamente sino al contrario deberá ser en forma gradual, para lo cual no se tratará solamente de introducir una innovación (CASE) ó una nueva metodología de desarrollo de sistemas, realmente se requerirá de muchos esfuerzos reconocidos ante todo, por el personal involucrado en el proceso de desarrollo del sistema.

10. Los nuevos sistemas: Si la calidad como se definió en un principio de esta tesis no es sólo satisfacer al cliente sino sobrepasar sus expectativas, debemos a través de la mejora continua lograr este objetivo desarrollando cada vez en menos tiempo y con más altos niveles de calidad y confiabilidad, siendo el elemento clave en esto la tecnología de información.

11. Promoción de la calidad: Dar a conocer continuamente a todo el personal la filosofía de calidad, los objetivos, la misión de la empresa, el plan y las políticas de calidad, todo esto para que se de un cambio de mentalidad.

- Retroalimentar los resultados a través de una publicación sobre calidad.
- Presentar los mejores sistemas en convenciones.
- Dar un reconocimiento a las mejores ideas.
- Llevar a cabo un intercambio de experiencias a través de reuniones(pueden formarse círculos de calidad).

12. Los recursos físicos: Para desarrollar un sistema de información con calidad requerirá invertir en tecnología de información como lo son las herramientas CASE así como en capacitación a todo su personal en metodologías y técnicas antes de aprender a utilizar estas herramientas. Además si desea tener una herramienta CASE poderosa que automatice todo el ciclo de vida del sistema o en dado caso diferentes herramientas modulares, será indispensable contar con un equipo computacional ya sea una estación de trabajo con un procesador 486 o *pentium* de ser posible y con suficiente capacidad en disco y de memoria (dependerá de la herramienta, pero considere que la mayoría consume muchos recursos), o una minicomputadora y de ser posible una herramienta que pueda trabajar en red siendo esto lo mejor, para el equipo de desarrollo de sistemas, ya que podrán compartir toda la información que se encuentre almacenada en el repositorio y podrá avanzarse más rápido al estar desarrollando el sistema.

13. Participación total: Todas las personas serán responsables de que se dé y se mejore la calidad, se requiere de un compromiso y de una participación de todos hacia ese nuevo reto sin fin.

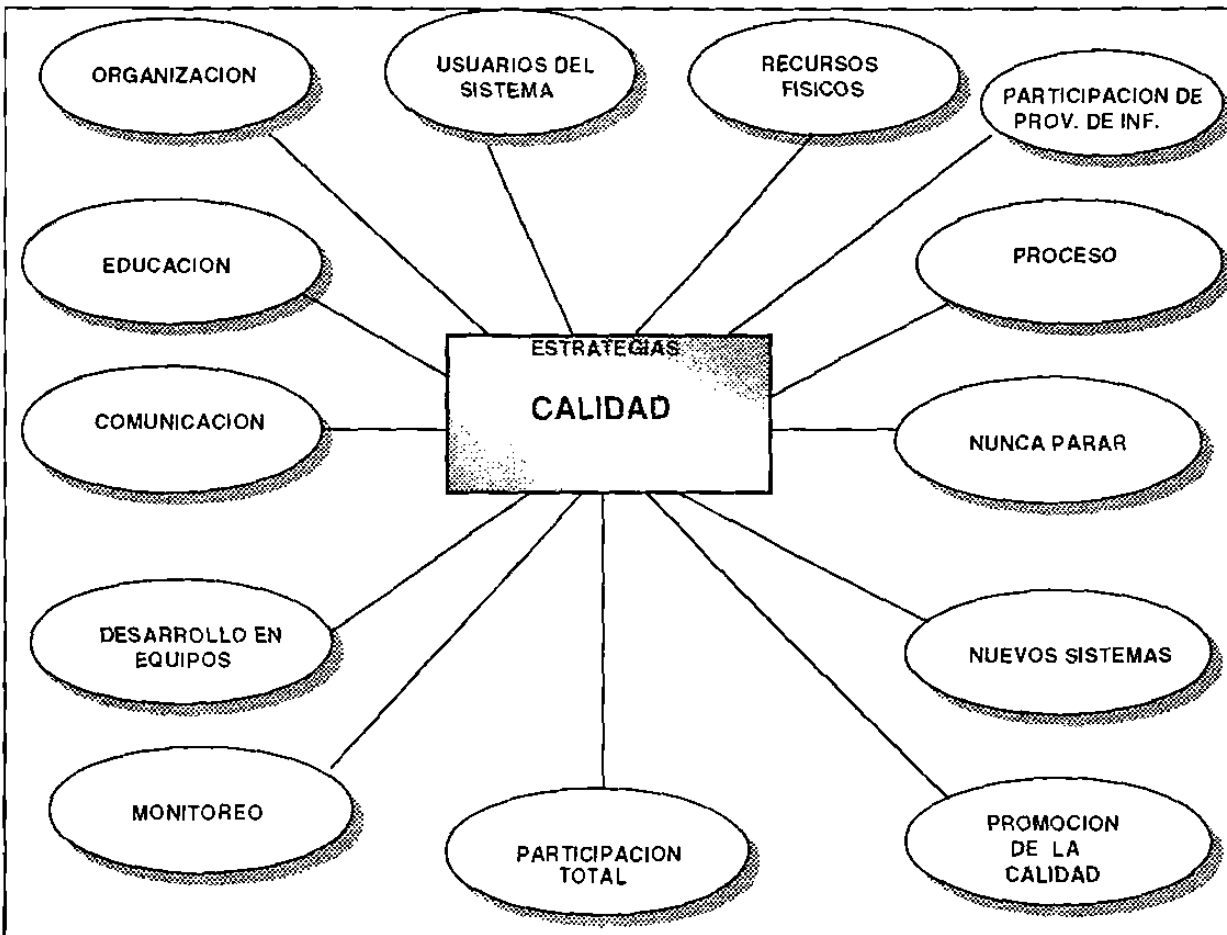


Fig.6.1 Estrategias para el logro de la calidad