

2947



# UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

SISTEMA PARA LA ADMINISTRACION DE UNA  
OBRA CIVIL Y SU CONTROL DE COSTOS  
EN EQUIPOS B-25

TESIS PROFESIONAL  
QUE PARA OBTENER EL TITULO DE  
INGENIERO EN COMPUTACION  
P R E S E N T A  
JAVIER VELAZQUEZ TAFOYA



Director de Tesis:  
ING. HECTOR CAMPOS ESTRADA

MEXICO, D. F.

1989



Universidad Nacional  
Autónoma de México



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

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

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

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

1. ANTECEDENTES	1
2. METODOLOGIAS DE OBRA CIVIL USADAS PARA EL DESARROLLO DEL SISTEMA ADMINISTRACION COSTO DE OBRA.	3
2.1 Metodología para el análisis de los Precios Unitarios	4
2.2 Metodología para la ruta crítica	8
3. METODOLOGIAS DE SISTEMAS SEGUIDAS PARA EL DESARROLLO DEL SISTEMA ADMINISTRACION COSTO DE OBRA	21
3.1 Metodología para el análisis de sistemas	22
3.1.1 Antecedentes	23
3.1.2 Objetivos y alcances	24
3.1.3 Descripción general del sistema	24
3.1.4 Descripción de entradas/salidas	26
3.1.5 Descripción funcional	28
3.1.6 Análisis de la información	31
3.1.7 Diccionario de datos	34
3.1.7.1 Estructuras de datos	34
3.1.7.2 Flujos de datos	35
3.1.8 Recursos y calendario	36
3.1.9 Anexos	36
3.2 Metodología para el diseño de sistemas	37
3.2.1 Descripción general	38
3.2.1.1 Principales componentes de "software"	38
3.2.1.2 Interfases de "software - hardware" y humanos	38
3.2.1.3 Limitaciones y alcances del diseño	39
3.2.2 Descripción de diseño	39
3.2.2.1 Revisión del D.F.D. (Diagrama de Flujo de Datos)	39
3.2.2.2 Revisión de estructuras de datos	40
3.2.2.3 Carta de estructura de "software"	40
3.2.3 Descripción de menús	42
3.2.3.1 Carta de estructura de menús	42
3.2.3.2 Descripción de cada menú	42
3.2.4 Sugerencias para el desarrollo de un buen "software"	43
3.3 Metodología para la implementación de sistemas	44
3.3.1 Características generales de la implementación	45
3.3.1.1 Elección del "software"	45
3.3.1.2 Tiempo de programación	45
3.3.1.3 Comentarios sobre la liberación	45
3.3.2 Probando el "software"	45
3.3.2.1 Pasos en las pruebas del "software"	46
3.3.2.1.1 Pruebas de unidad	46
3.3.2.1.2 Pruebas de integración	47
3.3.2.1.3 Pruebas de validación	47
3.3.2.1.4 Probando las funciones del sistema	48
3.3.2.2 El arte de la depuración	48
3.3.3 Mantenimiento del "software"	48
3.3.3.1 Mantenimiento mas fácil desde el desarrollo	48
3.3.3.2 Aspectos administrativos del mantenimiento del "software".	49

4. DESARROLLO DEL SISTEMA ADMINISTRACION COSTO DE OBRA	51
4.1 Análisis del sistema de Administración Costo de Obra.	52
4.1.1 Antecedentes del Sistema de Administración Costo de Obra	53
4.1.2 Objetivo a alcances del Sistema de Administración Costo de Obra	53
4.1.3 Descripción general del Sistema de Administración Costo de Obra	54
4.1.4 Descripción de entradas/salidas del Sistema de Admon Costo de Obra	59
4.1.5 Descripción funcional del Sistema de Administración Costo de Obra	63
4.1.6 Análisis de la información del Sistema de Administración Costo de Obra	67
4.1.7 Diccionario de datos del Sistema de Administración Costo de Obra	72
4.1.7.1 Estructuras de datos del Sistema de Administración Costo de Obra	72
4.1.7.2 Flujos de datos del Sistema de Administración Costo de Obra	75
4.1.8 Recursos y calendario para desarrollar el Sistema de Admon Costo de Obra	76
4.1.9 Anexos del Sistema de Administración Costo de Obra	76
4.2 Diseño del Sistema de Administración Costo de Obra	77
4.2.1 Descripción general del Sistema de Administración Costo de Obra	78
4.2.1.1 Principales componentes de "software" del sistema de Administración Costo de Obra	78
4.2.1.2 Intereses de "software-hardware" y humanos del Sistema de Admon Costo de Obra	79
4.2.1.3 Limitaciones y alcances del Sistema de Admon Costo de Obra	79
4.2.2 Descripción de diseño del Sistema de Administración Costo de Obra	80
4.2.2.1 Revisión del D.F.D. del Sistema de Administración Costo de Obra	80
4.2.2.2 Revisión de estructuras de datos del Sistema de Admon Costo de Obra	82
4.2.2.3 Carta de estructura de "software del Sistema de Administración Costo de Obra	90
4.2.3 Descripción de menús del Sistema de Administración Costo de Obra	99
4.2.3.1 Carta de estructura de menús del Sistema de Admon Costo de Obra	99
4.2.3.2 Descripción de cada menú del Sistema de Admon Costo de Obra	102
4.3 Implementación del Sistema de Administración Costo de Obra	105
4.3.1 Características generales de la implementación del Sist. de Admon Costo de Obra	106
4.3.1.1 Elección del "software" para el Sistema de Admon Costo de Obra	106
4.3.1.2 Tiempos de programación del Sistema de Admon Costo de Obra	106
4.3.1.3 Comentarios sobre la liberación del Sistema de Admon Costo de Obra	107
4.3.2 Probando el "software" del Sistema de Administración Costo de Obra	107
4.3.2.1 Pasos en las pruebas del "software" del Sistema de Admon Costo de Obra	107
4.3.2.1.1 Pruebas de unidad del Sistema de Administración Costo de Obra	107
4.3.2.1.2 Pruebas de integración del Sistema de Administración Costo de Obra	108
4.3.2.1.3 Pruebas de validación del Sistema de Administración Costo de Obra	108
4.3.2.1.4 Probando las funciones del Sistema de Administración Costo de Obra	108
4.3.3 Mantenimiento del "software" del Sistema de Administración Costo de Obra	108
4.3.3.1 Mantenimiento más fácil desde el desarrollo del Sist de Admon Costo	108
4.3.3.2 Aspectos administrativos del mantenimiento del Sist de Admon Costo	109
4.3.3.3 Problemas durante el mantenimiento del Sistema de Admon Costo de Obra	109
5. CONCLUSIONES	110
6. BIBLIOGRAFIA	112

## **1.-ANTECEDENTES**

**2.-METODOLOGIAS SEGUIDAS PARA EL  
DESARROLLO DEL SISTEMA  
ADMINISTRACION COSTO DE OBRA**

La organización de la obra, contando con el soporte técnico de la oficina central, el cual gravará a todas las obras de la empresa en un período determinado y considerando que cada obra tiene diferentes importes, tiempos de ejecución, localización, accesos, riesgos, personal técnico, personal administrativo, comunicaciones, fleles, oficinas de campo, almacenes, consumos, etc, a más de otros conceptos fuera de control de la empresa constructora y también variables tales como; gastos financieros por retraso en la tramitación y cobro de las estimaciones, escasez de materias primas imposibles de almacenar, retraso por mal tiempo, etc, consideramos injusto proponer condiciones promedio para todas las obras.

La estructura organizacional de la obra, es variable pero es necesario distinguir el área de producción y el área de control.

El costo de la oficina de obra. Para la valuación del costo de una organización de obra, podemos dividir sus gastos en cinco rubros principales que pueden ser:

- Gastos técnicos y/o administrativos.
- Traslado de personal.
- Comunicaciones y fleles.
- Construcciones provisionales.
- Consumo y varios.

Imprevistos de construcción. Consideramos indispensable precisar, que a cada nivel o etapa de un planteamiento económico, corresponde un imprevisto, cuando desafortunadamente se contrata un precio alzado sobre un anteproyecto, se confunde la indeterminación con los imprevistos de construcción.

En otras palabras, creemos que los imprevistos de construcción deben confinarse a aquellas acciones que quedan bajo el control y responsabilidad del constructor y que la provisión por indeterminaciones debe considerarse contingencia previsible y manejarse fuera del imprevisto y de la suma alzada.

Financiamiento. Antes y durante la ejecución de los trabajos de construcción, se efectúan fuertes erogaciones, es decir, cuando se excava el primer metro cúbico se ha hecho ya, una erogación considerable. La estricta vigilancia y supervisión de las inversiones en la obra, es, también requerimiento indispensable que obliga a esperar un lapso para cobrar la obra ejecutada, lo que convierte a la empresa en un financiero a corto plazo que forzosamente devenga intereses.

Utilidad. La utilidad en su concepción mas general, es a nuestro juicio, el objeto y la razón de toda obra ejecutada por el hombre. La obra inútil no tiene cabida en el mundo actual, donde necesitamos aprovechar al máximo todos los recursos disponibles y si en el pasado, no tuvo nunca justificación, en el presente, el desperdicio de recursos tanto materiales como humanos, es a nuestro juicio imperdonable.

Si analizamos en cualquier época las obras en la historia de la humanidad veremos que todas ellas cumplieron con un fin; desde el monumento cuya utilidad es esencialmente estética, hasta la primera fundición de acero que aún en forma rudimentaria inicia el cambio del destino del mundo, todas cumplieron con un fin determinado, que generó beneficios en algún sentido. Más aún, las obras actualmente denominadas de interés social, persiguen una utilidad a largo plazo.

Fianzas. El incumplimiento de las condiciones de un contrato implica un riesgo que la parte contratante evita por medio de fianzas y siendo éstas una erogación para la parte contratista, deben ser elementos del costo. La valuación de este cargo dependerá de las condiciones específicas y los requerimientos de la parte contratante. Existen siete principales tipos de fianzas, las cuales son:

- fianza de anticipo
- fianza de cumplimiento
- fianza para retirar el fondo de retención
- fianza de garantía de conservación
- fianza de pena convencional
- fianza de licitación
- fianza de anticipo ante el banco de obras.

Impuestos y derechos reflejables. Aquí se consideran los impuestos que la ley permite incluir en el costo.

**Federales.**

- Ingresos mercantiles, tasa general para constructoras.
- Prestaciones, derechos e impuestos sobre la mano de obra.

**Estatales y municipales**

- Tasa general para constructoras de ingresos mercantiles.
- Otros impuestos como, multas y recargos en el estado de México, sobre operaciones contractuales en el estado de Jalisco, etc.

**Impuestos especiales.**

- Secretaría del patrimonio nacional.
- Obras y servicios de beneficio regional.

Impuestos y derechos no reflejables. En este punto mencionaremos aquellos impuestos que la ley no permite incluir en el costo. Y por lo tanto afectarán (reduciendo) a la utilidad.

**Federales.**

- Impuesto sobre la renta.
- Aportaciones al fondo nacional de la vivienda.
- Participación de utilidades a los trabajadores.
- Pagos sobre dividendos.

**Estatales y municipales.**

- Dependerá de la legislación impositiva local.

**Impuestos especiales**

- Fomento para campos deportivos; ejidales.

A continuación intentaremos explicar, como se realiza el cálculo de los costos directos y cuáles son los factores que intervienen en este cálculo.

Hemos definido como Costo Directo a la suma de materiales, mano de obra y equipo necesarios, para la realización de un proceso productivo. Basándonos en esta definición podemos escribir un Costo Directo matemáticamente, de la siguiente manera:



$$[ax+by+cz+\dots+\&@] = C.D.$$

Considerando variables : x,y,z,...@

y variables condicionales : a,b,c,...&

Como variables podemos considerar el valor de los materiales, el valor de la mano de obra, y el valor del equipo; como variables condicionales podemos considerar las cantidades consumidas de cada uno de estos integrantes, es decir, la parte que representan dentro de un Costo Directo.

Podemos aceptar también que, las variables condicionadas pueden convertirse en constantes para una obra específica, o para un rango de obras promedio. Ahora bien, las variables de cantidades de materiales, de mano de obra y de equipo, también pueden ser constantes para un tiempo determinado. Resumiendo, "las variables lo serán en función del tiempo de aplicación", y "las variables condicionadas, lo serán en función del método constructivo, el tipo de edificación y de la tendencia estadística.

Si en un costo determinado llegásemos a convertir : "a", "b", "c", en constantes, determinadas por valores promedio estadísticos, tendríamos controlado una gran parte del proceso productivo y podríamos con mayor seguridad presuponer costos a tiempo inmediato y mediano, ya que, como su nombre lo indica, "presupuesto" no es una cosa que anticipar una serie de suposiciones con tendencias controladas a un tiempo inmediato. Cuando usamos la palabra antepresupuesto estamos queriendo decir con esto, que nuestras suposiciones son a un tiempo mediano y que necesitaremos revisarlas cuando sea necesario aplicarlas a un problema inmediato. Por lo tanto, el presupuesto ideal sería aquel que estuviere integrado por variables "controladas", que al serlo se convertirían en constantes.

### Especificaciones

Consideremos como definición de especificación a "la descripción detallada de características y condiciones mínimas de calidad que debe reunir un producto".

**Especificaciones generales.** En forma escrita y a manera de normas generales, existen una serie de agrupaciones que dictan especificaciones para cada una de las actividades especializadas, para el caso de la edificación podemos mencionar, el "Reglamento de construcciones para el distrito federal" del D.D.F., el "Reglamento de ingeniería sanitaria relativa a edificios", de la S.S.A., los reglamentos estatales y municipales, el "Instructivo para el diseño y ejecución de instalaciones de gas" de S.I.C., y en forma muy importante las normas de calidad de la dirección general de normas.

- 2.2.1.3 Dibujo de diagramas.
  - 2.2.1.3.1 Actividades reales.
  - 2.2.1.3.2 Actividades ficticias.
- 2.2.2 Programación
  - 2.2.2.1 Valuación de tiempos (tabla de tiempos).
    - 2.2.2.1.1 Jornadas.
    - 2.2.2.1.2 Días, horas, etc.
  - 2.2.2.2 Obtención ruta crítica (tabla de holguras).
    - 2.2.2.2.1 Fecha primaria.
      - 2.2.2.2.1.1. Iniciación.
      - 2.2.2.2.1.2 Terminación.
    - 2.2.2.2.2 Fecha última.
      - 2.2.2.2.2.1 Iniciación.
      - 2.2.2.2.2.2 Terminación.
  - 2.2.2.3 Análisis y reducciones.
    - 2.2.2.3.1 Holgura total.
    - 2.2.2.3.2 Holgura libre.
    - 2.2.2.3.3 Holgura independiente.
    - 2.2.2.3.4 Modificar secuencias.
    - 2.2.2.3.5 Modificar duración
- 2.2.3 Control.
  - 2.2.3.1 Uso de holguras (repartición de recursos).
  - 2.2.3.2 Reducción de tiempos.

Enunciados los tres pasos necesarios para realizar un proceso productivo, procederemos a analizar cada uno de ellos.

#### 2.2.1.1 Lista de actividades

En la lista de actividades no debemos olvidar los tiempos necesarios para proyectos y trámites, indispensables para la ejecución de un proceso productivo, así como las actividades específicas de construcción.

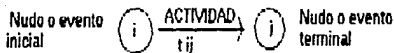
#### 2.2.1.2 Tabla de secuencias

En esta tabla de secuencias debemos señalar, cuáles actividades son simultáneas, cuáles inmediatas anteriores y cuáles inmediatas posteriores, tomando en cuenta indicar en dicha tabla de secuencias las limitaciones de espacio y de recursos, así como una muy importante, que es la decisión del responsable, es decir, la orden que aún no teniendo a primera vista razón lógica, debe efectuarse de ese modo. La lista de actividades y la tabla de secuencias se pueden reunir en una sola tabla como la que sigue:

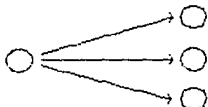
No.	Actividad	Anteriores	Posteriores

### 2.2.1.3. Dibujo de diagrama

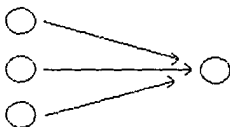
#### A) Representación de una actividad.



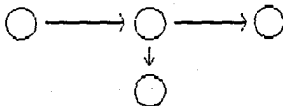
#### B) Fuente. Se define como un evento del cual parten varias actividades simultáneas



#### C) Resumidero. Se define como un evento al cual llegan varias actividades simultáneas



#### D) Las actividades siempre deben salir y deben llegar a eventos o nudos



#### E) Actividades de liga o ficticias. Se define como "actividad de liga" a aquella con duración cero, que nos indique la liga que existe entre dos eventos de manera que se permita la secuencia prevista.

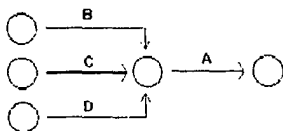
#### F) Representaciones gráficas de secuencia.

##### i) La actividad A es inmediatamente posterior a la actividad B



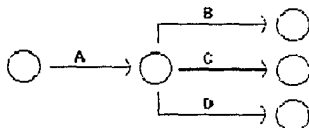
##### j) La actividad B es inmediatamente anterior a la actividad A

ii) La actividad A es inmediatamente posterior a las actividades B, C y D.



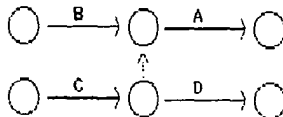
iii) Las actividades B, C y D, son inmediatas anteriores a la actividad A.

iiii) Las actividades B, C y D son inmediatas posteriores a la actividad A



v) La actividad A es inmediata anterior a las actividades B, C y D.

vi) La actividad A es inmediata posterior a las actividades B y C, y la actividad D sólo es inmediata posterior a la actividad C.



vii) La actividad C es inmediata anterior a la actividad D y las actividades B y C son inmediatas anteriores a la actividad A.

#### 2.2.2.1 Valuación de tiempo

Tomando en cuenta las limitaciones de espacio, de recursos y de decisión de reponsable, procedemos a hacer la valuación de los tiempos de duración de las actividades.

Es indudable que dependiendo del criterio personal del programador y del sistema de pago de la mano de obra, los tiempos de duración por actividad tengan un rango de variación muy grande. Siendo este tiempo decisión del programador, nosotros adoptaremos la siguiente fórmula para relacionarlo con el rendimiento y el volumen de obra a realizar.

---

No. de jornadas = Volumen de obra / Rendimiento de grupo

---

De donde la duración de cada actividad, dependerá del número de grupos que eficientemente puedan asignarse a la actividad en estudio, tomando muy en cuenta las limitaciones de espacio y de personal por lo cual, la "duración normal" de una actividad, será:

---

Duración normal = No. de jornadas / No. de grupos.

---

#### 2.2.2.2 Obtención de la ruta crítica

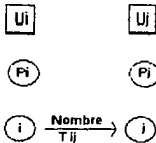
Como mencionamos anteriormente, la ruta crítica nos permite conocer las actividades que definen o determinan la duración de un proceso, es decir, las actividades críticas, ahora bien, existen otras actividades que no son críticas, y para llegar a su clasificación, necesitamos definir.

Holgura total, a la cantidad de tiempo que se puede retrasar una actividad sin afectar la terminación de un proceso.

Holgura libre, a la cantidad de tiempo que se puede retrasar una actividad sin afectar la fecha primera de iniciación de las posteriores.

Holgura independiente, a la cantidad de tiempo que se puede retrasar una actividad sin afectar la fecha última de las anteriores y la fecha primera de las posteriores.

Por consecuencia ruta crítica, también es una secuencia de actividades, cuya holgura total sea cero, ahora bien, si en la representación de una actividad cualquiera, tenemos un evento inicial, un evento terminal, una designación de actividad y un tiempo de duración, llamemos



Cuando más tarde se puede iniciar una actividad.

$U_i$

Cuando más tarde se puede terminar una actividad.

$U_j$

Cuando más pronto  
se puede iniciar  
una actividad.

$P_i$

Cuando más pronto  
se puede terminar  
una actividad.

$P_j$

y llamaremos  $t_{ij}$  el tiempo de realización normal de la actividad.

Por consiguiente, podemos decir que:

$$\text{Holgura total} = U_j - P_i - t_{ij}$$

$$\text{Holgura libre} = P_j - P_i - t_{ij}$$

$$\text{Holgura independiente} = P_j - U_i - t_{ij}$$

Para la tabulación de esta información nosotros recurrimos a la siguiente tabla:

-----  
Orden clave descripción actividad duración ip tp ir tr holgura  
-----  
-----

El proceso que se realiza para obtener los valores específicos en la ruta crítica, es el siguiente:

A. Tomar un listado de todas las actividades que van a conformar el programa, anotando las dependencias de cada actividad.

B. Obtener el ajuste cero de cada actividad, el cual es igual a el número de actividades que preceden a la actividad.

C. Ordenar las actividades, basándonos en los ajustes y las actividades posteriores.

D. Calcular la iniciación próxima (ip) y la terminación próxima basándonos en que,

$$\begin{aligned} ip &= tp \\ \text{actividad} & \text{ mayor de las actividades anteriores} \end{aligned}$$

$$\begin{aligned} tp &= ip + \text{duración} \\ \text{actividad} & \text{ actividad} \end{aligned}$$

E. Calcular la iniciación remota y la terminación remota, tomando en cuenta que,

$$\begin{aligned} tr &= ir \\ \text{actividad} & \text{ menor de las actividades posteriores} \end{aligned}$$

$$\begin{aligned} ir &= tp - \text{duración} \\ \text{actividad} & \text{ actividad} \end{aligned}$$

F. Calcular las holguras totales de la siguiente manera

$$\text{Holgura total} = t_r - t_p$$

### 2.2.2.3 Análisis y reducciones

Para analizar un proceso productivo, debemos de estudiar cada una de las holguras para reducirlas o conservarlas según convenga. Para las reducciones tenemos dos caminos, o modificamos las duraciones o modificamos las secuencias; la primera con más personal, con más equipo o con dobles turnos, y la segunda cambiando el sistema de programación.

#### 2.2.3.1 Repartición de recursos

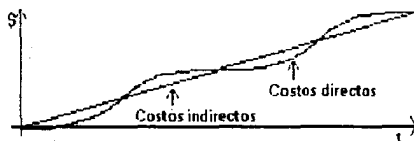
Si disponemos de holguras en las actividades, es lógico que podamos desplazar éstas, dentro de un programa de barras a su ubicación más conveniente y buscando que el personal no sea muy variable, o bien que las erogaciones no tengan máximos demasiado acusados.

Para dicha repartición se propone la siguiente tabla:

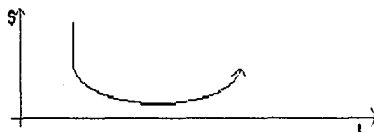
		SEMANAS					
		1	2	3			
i	j	descripción	grado	123456	123456	123456	
1a. Suma							
1a. Repartición							
2a. Repartición							

#### 2.2.3.2 Reducción de tiempos en función del costo

Todo proceso productivo produce dos clases de costos, costos indirectos y costos directos; siendo los primeros determinados por el funcionamiento de la empresa, y los segundos dependientes del tipo de obra de que se trate. Sabemos también que la mayoría de los costos fijos pueden representarse en la siguiente forma.



y los costos de obra se afectan al reducir tiempo de ejecución en la forma siguiente:



Siendo una gráfica de este tipo, la herramienta necesaria para que la persona encargada de la toma de decisiones, diga que tanto se incrementan los costos en función del tiempo.

**Ejemplo :**

Siguiendo la metodología propuesta anteriormente, realizaremos un ejemplo de ruta crítica.

A. tomar el listado de todas las actividades que van a conformar el programa , anotando las dependencias de cada actividad.

# secuencial	clave de actividad	actividad anterior	actividad posterior
1	A	P	L
2	B	X	N,S,U
3	C	P,J	L
4	D	S,U,Z	T
5	E	X	M,Z
6	F	T	P
7	G	N	R
8	H	V	L
9	J	T	C
10	K	-	X
11	L	A,C,H	-
12	M	E	U
13	N	B	G
14	P	F,Y	A,C
15	R	G	V
16	S	B	D
17	T	D	F,J,Y



18	U	M,B	D
19	V	R,W	H
20	W	X	V
21	X	K	B,E,W
22	Y	T	P
23	Z	E	D

B. Obtener el ajuste cero de cada actividad, el cual es igual a el número de actividades que preceden a la actividad

# secuencial	clave de actividad	actividad anterior	ajustes 0 1 2 3	actividad posterior
1	A	P	1	L
2	B	X	1	N,S,U
3	C	P,J	2	L
4	D	S,U,Z	3	T
5	E	X	1	M,Z
6	F	T	1	P
7	G	N	1	R
8	H	V	1	L
9	J	T	1	C
10	K	-	0	X
11	L	A,C,H	3	-
12	M	E	1	U
13	N	B	1	G
14	P	F,Y	2	A,C
15	R	G	1	V
16	S	B	1	D
17	T	D	1	F,J,Y
18	U	M,B	2	D
19	V	R,W	2	H
20	W	X	1	V
21	X	K	1	B,E,W
22	Y	T	1	P
23	Z	E	1	D

C. Ordenar las actividades, basándonos en los ajustes y en las actividades posteriores, de la siguiente manera:

Numeramos con uno a la actividad que tenga un valor cero en el renglón de ajustes, de ahí tomamos sus actividades posteriores y nos referimos a ellas posicionándonos en las mismas, restamos uno a el valor de ajuste de la actividad en la que estamos, si este valor de ajuste es igual a cero numeramos la actividad con el número secuencial correspondiente, una vez terminado este proceso tomamos la actividad numerada con dos y repetimos el proceso, así secuencialmente hasta terminar con todas las actividades.

número sec.	clave de actividad	actividad anterior	ajustes			actividad posterior	número orden.
			0	1	2 3		
1	A	P	1	0		L	21
2	B	X	1	0		N,S,U	3
3	C	P,J	2	1	0	L	22
4	D	S,U,Z	3	2	1 0	T	13
5	E	X	1	0		M,Z	4
6	F	T	1	0		P	17
7	G	N	1	0		R	10
8	H	V	1	0		L	16
9	J	T	1	0		C	18
10	K	-	0			X	1
11	L	A,C,H	3	2	1 0	-	23
12	M	E	1	0		U	8
13	N	B	1	0		G	6
14	P	F,Y	2	1	0	A,C	20
15	R	G	1	0		V	12
16	S	B	1	0		D	7
17	T	D	1	0		F,J,Y	15
18	U	M,B	2	1	0	D	11
19	V	R,W	2	1	0	H	14
20	W	X	1	0		V	5
21	X	K	1	0		B,E,W	2
22	Y	T	1	0		P	19
23	Z	E	1	0		D	9

Nuestra tabla ordenada y tomando en cuenta las duraciones de las actividades quedaria de la siguiente manera.

número ordenado	clave de actividad	actividad posterior	número orden.
1	A	L	21
2	B	N,S,U	3
3	C	L	22
4	D	T	13
5	E	M,Z	4
6	F	P	17
7	G	R	10
8	H	L	16
9	J	C	18
10	K	X	1
11	L	-	23
12	M	U	8
13	N	G	6
14	P	A,C	20
15	R	V	12
16	S	D	7
17	T	F,J,Y	15
18	U	D	11
19	V	H	14

20	W	V	5
21	X	B,E,W	2
22	Y	P	19
23	Z	D	9

D.- Calcular la iniciación próxima (ip) y la terminación próxima(tp), basándonos en que,

$$ip_{\text{actividad}} = tp_{\text{mayor de las actividades anteriores}}$$

$$tp_{\text{actividad}} = ip_{\text{actividad}} + \text{duración}$$

orden de cálculo	clave de actividad	actividad posterior	duración	valor de marcas (ip) (tp)	
1	K	X	24	0	24
2	X	B,E,W	12	24	36
3	B	N,S,U	24	36	60
4	E	M,Z	6	36	42
5	W	V	6	36	42
6	N	G	36	60	96
7	S	D	12	60	72
8	M	U	12	42	54
9	Z	D	18	42	60
10	G	R	12	96	108
11	U	D	24	60	84
12	R	V	6	108	114
13	D	T	60	84	144
14	V	H	12	114	126
15	T	F,J,Y	18	144	162
16	H	L	30	126	156
17	F	P	6	162	168
18	J	C	18	162	180
19	Y	P	12	162	174
20	P	A,C	18	174	192
21	A	L	6	192	198
22	C	L	12	192	204
23	L	-	6	204	210

E. Calcular la iniciación remota y la terminación remota, tomando en cuenta que,

$$tr_{\text{actividad}} = ir_{\text{menor de las actividades posteriores}}$$

$$ir_{\text{actividad}} = tp_{\text{actividad}} - \text{duración}$$

orden de cálculo	clave de actividad	duración	valor de marcas			(tr)
			(ip)	(tp)	(ir)	
1	K	24	0	24	0	24
2	X	12	24	36	24	36
3	B	24	36	60	36	60
4	E	6	36	42	42	48
5	W	6	36	42	156	162
6	N	36	60	96	108	144
7	S	12	60	72	72	84
8	M	12	42	54	48	60
9	Z	18	42	60	66	84
10	G	12	96	108	144	156
11	U	24	60	84	60	84
12	R	6	108	114	156	162
13	D	60	84	144	84	144
14	V	12	114	126	162	174
15	T	18	144	162	144	162
16	H	30	126	156	174	204
17	F	6	162	168	168	174
18	J	18	162	180	174	192
19	Y	12	162	174	162	174
20	P	16	174	192	174	192
21	A	6	192	198	198	204
22	C	12	192	204	192	204
23	L	6	204	210	204	210

F. Calcular las holguras totales de la siguiente manera

$$\text{Holgura total} = tr - tp$$

orden de cálculo	clave de actividad	duración	valor de marcas			Holgura	
			(ip)	(tp)	(ir)		
1	K	24	0	24	0	24	0
2	X	12	24	36	24	36	0
3	B	24	36	60	36	60	0
4	E	6	36	42	42	48	6
5	W	6	36	42	156	162	120
6	N	36	60	96	108	144	48
7	S	12	60	72	72	84	12
8	M	12	42	54	48	60	6
9	Z	18	42	60	66	84	24
10	G	12	96	108	144	156	48
11	U	24	60	84	60	84	0
12	R	6	108	114	156	162	48
13	D	60	84	144	84	144	0
14	V	12	114	126	162	174	48
15	T	18	144	162	144	162	0
16	H	30	126	156	174	204	48
17	F	6	162	168	168	174	6

18	J	18	162	180	174	192	12
19	Y	12	162	174	162	174	0
20	P	18	174	192	174	192	0
21	A	6	192	198	198	204	6
22	C	12	192	204	192	204	0
23	L	6	204	210	204	210	0

Una vez realizados todos los cálculos necesarios, se procede a realizar la gráfica de barras.

### **3. METODOLOGIAS DE SISTEMAS SEGUIDAS PARA EL DESARROLLO DEL SISTEMA ADMINISTRACION COSTO DE OBRA**

### **3.1 METODOLOGIA PARA EL ANALISIS DE SISTEMAS**

- 3.1.1 Antecedentes**
- 3.1.2 Objetivos y alcance**
- 3.1.3 Descripción general**
- 3.1.4 Descripción de entradas/salidas**
- 3.1.5 Descripción funcional**
- 3.1.6 Análisis de la información**
- 3.1.7 Diccionario de datos**
  - 3.1.7.1 Estructuras de datos**
  - 3.1.7.2 Flujos de datos**
- 3.1.8 Recursos y calendario**
- 3.1.9 Anexos**

Definición.- Obtener en forma detallada información del sistema, considerando: problemática, sistema actual, fuentes de información, y planear de manera formal las especificaciones de diseño.

### 3.1.1 Antecedentes

Es importante marcar que para poder realizar un análisis de cualquier sistema, previamente debemos haber hecho un estudio de factibilidad el cual debe contemplar:

- Los objetivos del sistema,
- estrategias de solución,
- resumen de costos,
- restricciones,
- descripción general del sistema actual,
- análisis costo-beneficio,
- evaluación técnica,
- conclusiones,

siendo este documento de tipo informativo dirigido a la gente que se dedica al aspecto administrativo y a los altos funcionarios.

Una vez realizado el documento del estudio de factibilidad, nos podemos basar en él para explicar a las personas técnicas (que es a quién se dirige un análisis del sistema) cuáles son los motivos que nos llevaron a realizar el presente documento de análisis y cuáles son los metas que se pretenden alcanzar.

Tanto el documento del estudio de factibilidad, como el análisis del sistema pueden ser entregados a las personas que manejan el lenguaje técnico, puesto que uno es el complemento de el otro. Pero no deben ser entregados ambos a un alto funcionario pues como el análisis es muy técnico solo causaría confusiones y entorpecería la toma de decisiones del funcionario, es recomendable que a este tipo de personas sólo se les presente un sencillo estudio de factibilidad.

A continuación presentamos los puntos que debe contener el documento de análisis, estos puntos son los siguientes:

- objetivos y alcances del sistema
- descripción general del sistema
- descripción de entradas/salidas
- descripción funcional
- diccionario de datos
- análisis de la información
- recursos y calendario
- anexos

Todos estos puntos serán explicados con mayor detalle a continuación.



### 3.1.2 Objetivos y alcances

Plantearemos aquí los reportes que el usuario desea manejar; tantos y cuantos sean solicitados deben ser presentados en este punto, para poder ser discutidos y analizados desde el punto de vista utilidad; siendo descartados aquí mismo todos aquellos que no muestren verdadera utilidad ó aquellos cuyo costo-beneficio es muy bajo.

De la misma forma, las consultas por pantalla que sean solicitadas se presentarán aquí; siguiendo con ellas, el mismo procedimiento que con los reportes.

Es necesario profundizar en las soluciones, para poder presentar en este punto detalladamente todas las formas de salida; este nivel de detalle debe especificar cuantos campos de captura va a tener cada pantalla, que tipo de ayuda se va a manejar, leyendas que contendrá cada salida a pantalla, etc. De la misma forma, leyendas que llevarán los reportes, formato que se utilizará en los mismos, etc.

Se debe de tener en cuenta que aunque el nivel al que se desea llegar es demasiado profundo, es mucho mejor desde un principio manejar la mayor cantidad de información posible, pues es muy común que como el usuario no está acostumbrado al manejo sistematizado de las cosas, quiera cambiar en el último momento uno o varios elementos dentro de la información, pero si nosotros lo obligamos desde un inicio a que nos defina toda la información que quiere manejar es más difícil que intente realizar cambios de último momento, es por esto que se insiste en la definición de los reportes y pantallas de salida con tanto énfasis.

El definir tan a detalle las salidas, no nos vuelve infalibles, es por esto que necesitamos estar concientes de que seguramente habrá cambios.

Por otro lado, se debe plantear por escrito, cuales son los objetivos y los alcances que no pueden estar definidos, ni dentro de una pantalla, ni dentro de un reporte, como por ejemplo el hecho de facilitar la toma de decisiones de una determinada persona o funcionario no puede estar plasmada en un reporte, pero es muy importante hacerlo notar puesto que en gran número los sistemas son desarrollados para proporcionar este tipo de ayuda.

### 3.1.3 Descripción general del sistema

Es importante describir en forma general, como estará conformado nuestro sistema global; para poder realizar esta descripción manejaremos dos formas: la primera forma para realizar la descripción general del sistema es una narrativa sencilla y con todos los tecnicismos necesarios en la cual plantearemos procesos, entradas y salidas a los mismos, esta descripción del proceso solo contendrá una explicación referente a la función de cada parte del sistema. Para saber como dividir estos procesos podemos tomar en cuenta las funciones que deben realizar, los valores que deben entrar y los resultados que deben entrar esto es referente a que si por ejemplo por un lado se realiza el cálculo de la desviación estándar de un determinado grupo de valores, y por otro se realiza el cálculo de la suma total de estos valores, todo esto lo conjuntaremos en un solo módulo llamado o identificado como cálculos.

En cuanto a las entradas y salidas de los módulos es conveniente definirlos tanto en nombre, contenido y dirección. El nombre es recomendable que se asigne acorde a el contenido, el contenido deberá llevar tipo y tamaño y la dirección nos indicará el módulo del cual sale y hacia el cual entra.

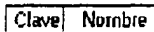
La 2a. forma para realizar la descripción general del sistema consiste en lo que conocemos como un diagrama de flujo de datos, el cual, tiene como objetivo el desarrollar gráficamente la transformación sucesiva de datos a lo largo de un conjunto de procesos. Para el manejo del diagrama de flujo de datos utilizaremos la siguiente simbología:



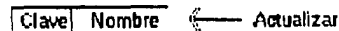
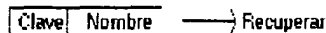
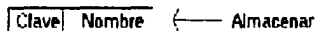
**Proceso.** - Convierte un flujo o flujos de entrada a un flujo o flujos de salida, el proceso se presenta como un círculo y considera los siguientes elementos: clave, que es un código asignado al proceso; nombre, debe representar de la manera más clara posible lo que hace el proceso; área, lugar en donde se efectúa el proceso.

**Nombre del paquete**  
 .....>

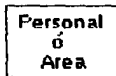
**Flujo.** - Un flujo de datos representa un paquete en movimiento; un paquete puede ser un dato, un conjunto de datos o materiales; se encuentra representado por una flecha que indica el sentido del movimiento.



**Almacenamiento.** - Un almacenamiento es un conjunto de paquetes estáticos en el que es posible; almacenar, actualizar, eliminar y recuperar; representaremos gráficamente a estos puntos de la siguiente manera.



Entidades externas.- Hogar en el que se inicia o finaliza un proceso o un conjunto de procesos en un sistema dado, pueden existir más de una entidad externa y se representa de la siguiente manera:



Los diagramas de flujo de datos deben cumplir con las siguientes características:

- Todo diagrama debe tener un inicio y/o final en una entidad externa o en un almacenamiento.
- No puede existir un proceso que contenga únicamente salidas.
- No puede existir un proceso que contenga únicamente entradas.
- Un diagrama de flujo de datos debe estar bien especificado en cuanto a nombres y claves.

Para poder realizar un diagrama de flujo de datos eficiente y confiable se recomienda seguir los pasos a continuación mencionados:

- Determinar las entidades externas del proceso.
- Definir el tipo de entradas y salidas que se generan hacia las entidades externas.
- De acuerdo a los dos puntos anteriores se inicia el dibujo de un diagrama, iniciando la secuencia de procesos de las entradas hacia las salidas definidas.

Los pasos anteriormente mencionados son para un primer diagrama o diagrama de prueba, una vez revisado y corregido se enuncian a continuación los pasos complementarios para obtener el diagrama de flujo de datos final.

- Detallar en un segundo diagrama de flujo de datos los procesos definidos.
- El diagrama anterior se complementa con el manejo de errores y excepciones.
- Realizar una revisión con el usuario para corregir posibles errores u omisiones.

### 3.1.4 Descripción de entradas/salidas

Debemos describir en forma detallada todas las pantallas de captura, los formatos de entrada y los formatos de salida. Los pantalles de captura deberán llevar nombre del campo en el que se va a capturar, leyenda que llevará antes del campo a capturar y tamaño el campo. Esto para todos los campos que van dentro de una pantalla agrupados con el nombre de la pantalla y la librería en la que estará; por ejemplo:

Librería: Formas.lib

Forma: Pantalla 1.form

Campos:

<u>Nombre del campo</u>	<u>Leyenda</u>	<u>Tamaño</u>
Clave	Clave del alumno	9
Nombre	Nombre del alumno	50
Calific 1	1a. calificación	2 . 2
Calific 2	2a. calificación	2 . 2
Calific 3	3a. calificación	2 . 2

Seguido de la pantalla se podrá poner un comentario explicativo del porque de la pantalla y sus características.

Para el manejo de las salidas necesitaremos saber la información que llevará cada una de ellas, ayudándonos de los siguientes formatos, para emitir la información conveniente, se deberá escoger alguno de los dos formatos.

#### 1er. Formato

---

Clave de salida:

Proceso generador:

---

<u>Datos (s)</u> <u>de salida</u>	<u>Nombre (s) de el(los)</u> <u>datos (s) en el</u> <u>diccionario de datos</u>	<u>Generado</u> <u>¿si/no?</u>	<u>Forma de</u> <u>generación</u>
--------------------------------------	---	-----------------------------------	--------------------------------------

---

**Clave de salida.** - La clave de salida es una clave que identifica a este reporte en particular.

**Proceso generador.** - Estas formas de salidas estan ligadas a nuestro diagrama de flujo de datos (expuesto en el punto anterior) de tal manera que aqui indicaremos el proceso que genera este reporte con el "nombre" obtenido del diagrama de flujo de datos.

**Dato(s) de salida.** - Indicará cuántos y cuáles son los datos que llevará este reporte identificados por un nombre que los describa lo mejor posible.

**Nombre del dato en diccionario de datos.** - El diccionario de datos es una parte importante dentro del análisis, por lo tanto lo discutiremos en puntos posteriores, solamente podemos mencionar que el nombre que aquí pondremos debe ser tomado de este diccionario de datos.

**Generado si/no?.** - Indicaremos aquí si este dato fué generado o si es un dato capturado o fijo para conocer así su fuente.

**Forma de generación.** - Si nuestro dato fué generado, debemos decir cual fué el proceso que lo generó y de ser posible dar una breve explicación de como fué generado.

## 2º Formato.

---

Sistema:

---

Formato de salida

---

Dispositivo:

Periodicidad:

Volumen:

Tipo de proceso:

Tiempo de respuesta:

Proceso generador:

---

**Sistema.**- Nombre del sistema sobre el cuál estamos trabajando, se puede hacer uso de siglas o alguna clave predefinida.

**Dispositivo.**- Nombre del dispositivo en el que se generará el reporte.

**Volumen.**- Se especificará el número aproximado de líneas o de hojas que serán emitidas.

**Tiempo de respuesta.**- El tiempo de respuesta es un parámetro que deberá ser llenado en cuanto se generen los primeros resultados para tener un punto de comparación en caso de posibles mejoras.

**Periodicidad.**- Tomando en cuenta que un reporte de emisión semestral o anual es un tanto ineficiente por su periodicidad este parámetro debe ser llenado con la mayor veracidad posible para poder definir si el reporte es viable o no.

**Tipo de proceso.**- Existirán dos tipos de procesos generadores, los cuales son: proceso en batch o proceso en línea, indicaremos aquí entonces cual de estos tipos de proceso generan el reporte.

**Proceso generador.**- Por último se indica cuál es el nombre del proceso que genera nuestro reporte, este nombre debe ser congruente con el asignado en el diagrama de flujo de datos.

### 3.1.5 Descripción funcional

Para poder realizar una descripción funcional de nuestro sistema, debemos recurrir a lo que anteriormente definimos como diagrama de flujo de datos, en el cuál especificamos los procesos que se deben de realizar, cada uno de estos procesos están formados por funciones, las cuales aquí deberán ser descritas; resumiendo, en este punto debemos describir a detalle cuáles son las funciones de cada uno de los procesos.

Para poder definir cuál es el mejor camino para esta descripción de funciones nos hemos basado en la siguiente tabla (2) :

Herramienta uso	Diagrama de flujo	Pseudo- código	Narrativa	Arboles de decisión	Tablas de decisión
Verificación de lógica	Bueno	Moderado	Muy malo	Moderado	Bueno
Estructuras lógicas	Malo	Bueno	Muy malo	Bueno	Bueno
Verificación por el usuario	Moderado	Moderado	Malo	Moderado	Malo
Especificación para el diseño	Moderado	Muy bueno	Malo	Malo	Moderado
Fácilmente implementable	Moderado	Bueno	Malo	Moderado	Moderado
Transporta- bilidad	Moderado	Bueno	Malo	Bueno	Moderado

Es por esto que se decidió que la mejor forma para hacer una descripción es el pseudocódigo, siendo este una herramienta muy potente.

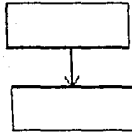
el objetivo del pseudocódigo es describir los procesos tratando de considerar una especificación completa para el diseño y la programación; eliminando diagramas de flujo y procurando ser fácilmente actualizable.

Es recomendable, para realizar un buen grupo de pseudocódigos, utilizar las siguientes siete recomendaciones (3) :

1º La lógica de los procesos se expresa como una combinación de las estructuras lógicas de control:

- Secuenciación
- Condicional
- Repetitiva
- Selectiva.

Secuenciación. - Cada bloque corresponde a una sección del pseudocódigo, se ejecutan en secuencia, se pueden agrupar en conjuntos bajo un solo nombre



**Condiciona**l.- Ejecución de uno de dos procesos dependiendo de alguna condición, uno de los procesos puede estar vacío ó el proceso a realizar puede ser la unión de una o varias estructuras lógicas.

**SI (condición) ENTONCES**  
 <estructuras lógicas>  
**DE OTRA MANERA**  
 <estructuras lógicas>.

**Repetivas**.- Existen dos formas de estructuras repetivas.

La primera forma es.- Ejecución de un proceso varias veces hasta que se cumpla una condición, si la condición se cumple la primera vez, no se ejecuta el proceso ni una sola vez.

**MIENTRAS (condición)**  
 <estructuras lógicas>  
 (fin del mientras)

La segunda forma es.- Ejecución de un proceso en forma repetiva, si se cumple una condición, la condición se evalúa después de ejecutar el proceso.

**REPITE**  
 <estructuras lógicas>  
**HASTA (condición)**

**Selectivas**.- Ejecución de una o varias estructuras lógicas dependiendo de alguna condición, una o varias de las estructuras lógicas pueden estar vacías.

**Selecciona (Variable)**

1: <estructuras lógicas>  
 2: <estructuras lógicas>  
 .  
 .  
 .

(fin de selección)

2\* Cuando una palabra o dato se encuentre referido al diccionario de datos debe subrayarse, este diccionario de datos será especificado posteriormente con el título "Diccionario de datos" pero es necesario que exista una relación en el nombre usado aquí y el nombre usado en el diccionario de datos.

3º Deben definirse los procesos en una forma jerárquica, por ejemplo:

- 1
- 1.1
- 1.2
- 1.2.1
- 1.2.2

para que en caso de tener un pseudocódigo un poco largo se puede dividir en módulos.

4º No se debe utilizar para la descripción de un proceso más de tres hojas.

5º Se puede tener una descripción muy detallada (a nivel instrucción de un programa) eliminando las reglas de sintaxis de un lenguaje formal.

6º Son válidas todas las palabras del español, y lo único que requiere es ajustarse a las estructuras de control.

7º Para referirnos a una subrutina podemos escribir el título dentro del pseudocódigo con mayúsculas.

### 3.1.6 Análisis de la información

Una vez que nosotros tenemos definida toda la información que se utilizará; debemos comenzar a trabajarla, para mejorar la eficiencia de los procesos en los que esta información será procesada, para esto existe una técnica conocida por todos nosotros la cual se conoce como normalización de archivos; técnica que explicaremos a continuación.

Entendemos por normalización la definición más sencilla de la información. Nosotros manejamos tres formas normales, pues consideramos que con estas se tiene la información en su forma más sencilla (4).

Un elemento que es muy importante para los diseñadores de bases de datos es saber como se van a agrupar los campos que forman parte de las tablas. Existen miles o cientos de formas para agrupar estos campos, pero algunas de estas formas son mejores que otras.

La primera forma normal convierte una tabla no simple en dos tablas simples. Esto es, en caso de encontrar dentro de nuestra tabla vectores o matrices, estos vectores o matrices deben de salir de nuestra tabla inicial, para formar parte de una segunda tabla dejando solamente una liga entre ambas tablas.

El siguiente proceso de normalización examina la primera forma normal y revisa la relación existente entre el campo llave y los demás campos, los campos deben depender directamente de la llave, en caso de no existir dicha dependencia los campos deben salir de la tabla, el resultado de este proceso lo conocemos como segunda forma normal. El autor Jean Martin en su libro titulado "Computer data-base organization" se expresa de la segunda forma normal de la siguiente manera, " todos los campos no llave deben tener una dependencia funcional completa con respecto al campo asignado como



llave", entendemos por dependencia funcional completa cuando la función de un campo depende de la función de otro campo pero no es subconjunto de este.

El paso final es llegar a la tercera forma normal, para lograr esto revisaremos que no exista dependencia funcional entre los campos que no son llave, la dependencia funcional la podemos explicar de la siguiente manera. Sean dos campos A y B, decir que B es funcionalmente dependiente de A equivale a decir que A identifica a B. En otras palabras, si en algún momento el valor de A es conocido entonces el valor de B se puede determinar.

Para poder entender mejor el proceso de la normalización daremos un ejemplo ilustrativo.

Nombre del archivo: STAFF  
Elementos: Idstaff\*  
Nombre  
Código-proyecto  
Código-cliente  
Historia-proyectos (n ocurrencias)  
Cod-proyecto  
Fecha-inicial  
Fecha-final  
Puesto  
Descripción-proyecto

#### 1a. Forma normal

Nombre del archivo: STAFF 1  
Elementos: Idstaff\*  
Apellido-paterno  
Apellido-materno  
Nombre  
Código-proyecto  
Código-cliente  
Nombre del archivo: STAFF 2  
Elementos: Idstaff-cod-proyecto  
Fecha inicial  
Fecha final  
Puesto  
Descripción-proyecto

#### Ventajas

- 1° El atributo nombre se encuentra disponible en sus elementos básicos, facilitando búsquedas.
- 2° Se eliminaron posibles problemas de actualización y de búsqueda.
- 3° Se eliminó la posibilidad de saturación del vector.
- 4° Se optimizó el espacio asignado.

#### 2a. Forma normal

Nombre del archivo: STAFF 1  
Elementos: Idstaff\*

Apellido-paterno  
Apellido-materno  
Nombre(s)  
Código-cliente  
Código-proyecto

Nombre del archivo: STAFF 2  
Elementos: Idstaff-cod-proyecto\*  
Fecha-inicial  
Fecha-final  
Puesto

Nombre del archivo: STAFF 3  
Elementos: Cod-proyecto\*  
Descripción-proyecto

**Ventajas.**

- 1° Se evitan errores de actualización.
- 2° Facilidad de actualización.
- 3 Evitar redundancias.

**3a. Forma normal**

Nombre del archivo: STAFF 1  
Elementos: Idstaff\*  
Apellido-paterno  
Apellido-materno  
Nombre(s)  
Código-proyecto

Nombre del archivo: STAFF 1  
Elementos: Cod-proyecto\*  
Cod-cliente

Nombre del archivo: STAFF 2  
Elementos: Idstaff-cod-proyecto\*  
Fecha-inicial  
Fecha-final  
Puesto

Nombre del archivo: STAFF 3  
Elementos: Cod-proyecto\*  
Descripción-proyecto

**Ventajas**

- 1° Se evitan errores de actualización
- 2° Facilidad de actualización
- 3° Evitar redundancias

### 3.1.7 Diccionario de datos

Dentro de este punto y para poder definirlo detalladamente tenemos que dividirlo en dos partes, lo que sería las estructuras de datos y el flujo que siguen estos datos.

#### 3.1.7.1 Estructura de datos

En esta sección debemos definir como estarán integrados nuestros datos, refiriéndonos a definición y organización de las tablas que forman nuestra base de datos o de los archivos que conforman el conjunto de la información a utilizar, es conveniente para el manejo de los datos seguir algunas especificaciones preestablecidas; en este caso para el manejo de la información se siguió la notación de "Yourdon" (5) para la definición de datos puesto que esta cumple con la mayoría de los requerimientos buscados por nosotros, además de ser fácilmente entendible; a continuación presentamos sus características generales:

Simbolo	Significado
=	El dato de la izquierda está compuesto por los elementos de la derecha
+	Unión de dos o más datos
{ }	Es un dato opcional
A/B	Seleccionamos el dato A o el dato B según convenga
**	Principio y fin de comentarios
{A}	El dato A es iterativo

Siguiendo la notación establecida pondremos el ejemplo de como está conformado un registro del archivo AUTORES.

reg-autores= Clave-autor + nombre- + institución + ((Instituciones-secundarias)) +  
domicilio-cases/domicilio-oficina \*\* Solo un máximo de 3  
Instituciones- secundarias\*\*

Además deberemos tener un formato en el que presentemos las características particulares de las tablas de nuestra base de datos o de los archivos que conformen el sistema. Siguiendo con nuestro ejemplo lo llenaremos de la siguiente manera:

Flujo de datos			
Descripción General del Sistema de los datos de la			
No	Descripción	Tipo	Observaciones
1	Clave del autor	char (5)	
2	Nombre del autor	char (50)	
3	Clave de la instit.	char (5)	máximo tres
Organización física:		Volumen esperado:	
llave campo 1		se desconoce	

En caso de necesitar algunas aclaraciones se pueden escribir estas al pie de la forma anterior.

### 3.1.7.2 Flujo de datos

Aquí debemos dar a conocer cual es el flujo de los datos, cabe señalar que este flujo de datos es referido al diagrama de flujo de datos discutido en el punto 3.1.3 titulado " Descripción General del Sistema " y que, no es posible tener ningún tipo de excepción, esto es que todos los datos que aquí manejamos deben estar incluidos en alguna parte de nuestro diagrama de flujo de datos.

Para manejar nuestro flujo de datos utilizaremos la siguientes forma:

Flujo de datos			
Fuente :			
Destino :			
No	Descripción	Tipo	Observaciones
Volumen :			

El formato anterior se llenará de la siguiente manera:

Flujo de datos: Daremos algún nombre a los datos que se manejarán en esta forma.  
Fuente.- Nombre del proceso del cual salen nuestros datos, proceso del diagrama de flujo de datos (D.F.D)  
Destino.- Nombre del proceso al cual llegan los datos, dentro del (D.F.D)

Los siguientes cuatro campos tienen la característica de poder ser varios, esto es cuando diferentes datos deben de salir todos del mismo proceso y deben llegar todos a otro proceso.

No. - Número secuencial de los grupos de datos.

Descripción. - Leyenda para saber que se maneja.

Tipo. - Archivo, carácter, valor entero, etc.

Observaciones. - Descripción de características especiales.

El volumen que es el campo hasta abajo de la forma debe de calcularse como transacciones esperadas/tiempo.

### 3.1.8 Recursos y calendario

Hasta el punto anterior damos por terminado nuestro análisis como tal, pero es importante tener un punto de comparación y datos suficientes para poder realizar un cobro en los casos necesarios, es por esto que se debe especificar, el número de personas que se necesitaron para realizar el presente análisis, el número de horas que cada una de estas personas dedicaron al presente trabajo, los gastos externos que implicó el análisis como pudieran ser, viáticos, comidas, hospedajes, etc., y así todos los recursos que se debieron invertir para obtener un resultado satisfactorio.

Pero así como es necesario conocer los recursos también es importante saber las fechas estimadas y las fechas reales que se dieron en el presente trabajo.

### 3.1.9 Anexos

Por último es conveniente tener una historia del tipo de manejo que se llevaba de la información, así como las causas que originaron el presente trabajo, estos dos puntos es necesario que sean a detalle puesto que es el sustento de un futuro sistema a desarrollar.

## **3.2. METODOLOGIA PARA EL DISEÑO DE SISTEMAS**

- 3.2.1 Descripción general**
  - 3.2.1.1 Principales componentes de "software"**
  - 3.2.1.2 Interfases de "software - hardware" y humanos**
  - 3.2.1.3 Limitaciones y alcances del diseño**
- 3.2.2 Descripción de diseño**
  - 3.2.2.1 Revisión del D.F.D. (Diagrama de Flujo de Datos)**
  - 3.2.2.2 Revisión de estructuras de datos**
  - 3.2.2.3 Carta de estructura de "software"**
- 3.2.3 Descripción de menús**
  - 3.2.3.1 Carta de estructura de menús**
  - 3.2.3.2 Descripción de cada menú**
- 3.2.4 Sugerencias para el desarrollo de un buen "software"**

Definición.- El diseño de los sistemas es una especificación clara y completa de los requerimientos de "software" de un sistema.

### 3.2.1 Descripción General

#### 3.2.1.1. Principales componentes de "software"

Expondremos aquí las características que posee el equipo que vamos a utilizar para el desarrollo del sistema en cuanto a "software"; deberemos exponer el sistema operativo que maneja, las características del mismo y de ser posible un brevuario con los comandos más usados y su forma de manejo. Además de puntualizar cada uno de los lenguajes que maneja, junto con las ventajas y las desventajas de cada uno de estos lenguajes enfocadas al sistema que vamos a desarrollar. Será necesario también una explicación de los paquetes complementarios del equipo como lo son: hoja de cálculo, procesador de palabras, paquete de gráficos, etc.; por último en caso de poseer una base de datos esta también se deberá mencionar, junto con el manejador de la base de datos y las características del mismo.

Todo lo expuesto anteriormente nos ayudará a tomar una decisión en cuanto al tipo de "software" que utilizaremos para la implementación de nuestro sistema, siendo esta una decisión muy importante, es necesario que las especificaciones de los alcances de cada software en particular (lenguajes, paquetes, manejadores de base de datos) sean muy detalladas para definir con que herramientas se cuentan y tomar así el camino correcto en la decisión del software.

#### 3.2.1.2. Interfases de "software-hardware" y humanos

La comunicación que existirá entre el "software" que se planea desarrollar y las personas que lo utilizarán debe ser tomada muy en cuenta, puesto que en la actualidad el rechazo hacia los sistemas computarizados es muy grande y es mayor en tanto menor sea la capacidad de la persona que utilizará estos sistemas; es por esto que es necesario plantear en este punto diferentes tipos de interfases entre el "software" y el usuario, tomando de estos diferentes tipos el que proporcione menor complejidad y mayor cantidad de ayudas en línea; nosotros recomendamos utilizar en el 100% de los sistemas el manejo de menús puesto que de esta forma se va dosificando la información y es más fácil de comprenderla, por otro lado es muy importante utilizar ayudas en línea, verías en cada menú pues es muy normal que una persona sin experiencia en el medio no sepa que acción tomar si no se le especifica. Es recomendable que el "software" desarrollado sea vistoso, como por ejemplo recuadros parpadeantes o de colores vivos en caso de error y cosas de este tipo, para captar la atención del usuario. La última recomendación en cuanto a las interfases "software"-humanas es que absolutamente todos los campos de captura sean validados, puesto que una vez liberado un sistema la mayoría de los problemas son en la información, este parte de validación es el desarrollo del sistema pero si tomamos en cuenta los beneficios que nos entregará es muy recomendable realizarlo.

En cuanto a la comunicación que existirá entre el "hardware" y los humanos, cabe señalar que existe un "hardware" específico para aplicaciones específicas, nos referimos al uso de terminales, impresoras de validación, cajas registradoras, lectores ópticos, etc., el "hardware" que aquí mencionamos no es el disco ni el CPU de la microcomputadora, sino el "hardware" que esta en contacto con el usuario.

Aquí debemos pensar en "hardware" específico, para una aplicación en particular, si por ejemplo nuestro sistema es para dar solución a un problema existente en la lesorería relacionado con el cobro de impuestos, es mucho más conveniente ponerle al cobrador una caja registradora, con un lector óptico y una impresora de validación que ponerle una terminal de algún equipo grande como A-9 o de algún minicomputador. Por el contrario si la solución a nuestro sistema es la captura de datos de algunos formatos prellenados; la caja registradora no es una óptima solución.

Con lo anterior queremos explicar, que podemos definir un "hardware" perfecto o casi perfecto para cada problema diferente que se nos presente.

### 3.2.1.3. Limitaciones y alcances del diseño

Debemos definir en este punto que tan ambicioso será nuestro plan de "software"; es decir, una explicación de todos los puntos que se intentarán abarcar. Al mismo tiempo debemos explicar las limitaciones encontradas por las cuales no se pueda dar solución a algún problema en específico de nuestro gran problema.

Por ejemplo, se nos solicita realizar la toma de datos y el análisis de los mismos, referentes a los sismos de 1990 en adelante.

Aquí podremos definir diferentes alcances, como son:

- Cálculo de media, variancia, desviación estándar, etc.
- gráficas de barras de los sismos.
- etc.

y también podemos definir las limitaciones con las que nos encontramos, un ejemplo de limitación pudiera ser que no se cuenta con un "plotter", si se contara con él se podrían entregar una mayor cantidad de gráficas y de mucha mayor calidad.

## 3.2.2 Descripción de diseño.

### 3.2.2.1 Revisión del diagrama de flujo de datos.

Nos referimos al punto 3.1.3 titulado "Descripción General", dentro de este punto debemos haber hecho un diagrama de flujo de datos (D.F.D) inicial; para estas etapas del diseño es muy factible que varios elementos de nuestro D.F.D. necesiten ser cambiados de lugar o eliminados, así como que algunos que no existan se piensen en datos de allá.

Como se comentó en el documento de análisis el usuario con poca experiencia en la mecanización de las cosas no toma en un principio en cuenta todos los elementos que estarán dentro del sistema y si a eso agregamos un análisis con poca experiencia, el resultado es que el diagrama de flujo de datos planteado en el principio debe tener una gran cantidad de cambios, es por esto que en este punto sugerimos realizar una revisión detallada del diagrama de flujo de datos, dejando como resultado un D.F.D. final al cual será la base o sustento principal de nuestro sistema.

Este punto es opcional; si la persona que realizó el análisis es una persona con mucha experiencia en el campo, tal vez considere que este punto es redundante, es por eso que lo dejamos a su consideración el realizar o no la revisión del D.F.D.



### 3.2.2.2 Revisión de estructuras de datos.

En esta parte tomaremos en cuenta dos puntos desarrollados en el capítulo de "Metodología para el Análisis de Sistemas", los cuales son el 3.1.7.1 denominado "Estructuras de Datos" y el punto 3.1.6 denominado "Análisis de la Información", dentro de la estructura de datos especificamos como se debían presentar los datos que conforman nuestra información, y dentro del punto llamado análisis de la información se debió realizar la normalización de estos datos, aquí juntaremos el resultado obtenido de la normalización con los formatos de presentación de esta información, y entregaremos como resultado en el formato propuesto para la presentación de las estructuras de datos, los archivos normalizados y corregidos.

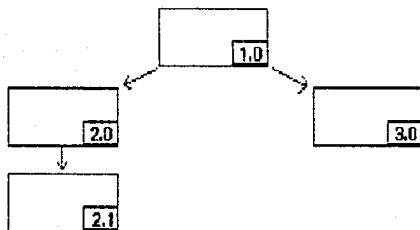
Este punto nos servirá además para revisar nuevamente si las normalizaciones hechas están correctas, aquí no podemos dejarlo como punto opcional puesto que entregaremos un nuevo y último formato de nuestros datos.

En este momento realizaremos la última modificación a las estructuras de datos, en puntos posteriores no es válido realizar modificaciones a estas estructuras.

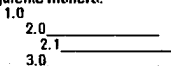
### 3.2.2.3 Carta de estructura de "software"

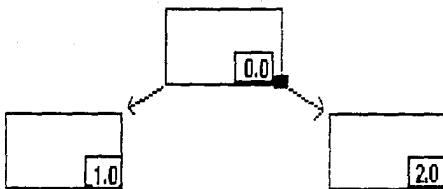
Nos apoyaremos en una carta estructurada, para desarrollar los elementos que componen cada uno de los procesos definidos dentro de nuestro diagrama de flujo de datos.

En el desarrollo de esta carta de estructura utilizaremos la siguiente simbología



Como podemos observar en el diagrama anterior, la estructura de "software" muestra la dependencia jerárquica. El diagrama anterior también se puede entender de la siguiente manera.

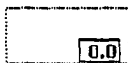




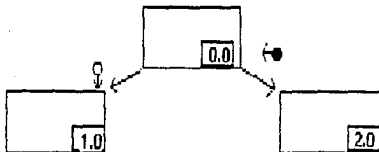
En el diagrama anterior, el cuadro en el módulo cero nos indica un condicionamiento para el llamado del módulo dos.



En ocasiones tendremos módulos iterativos (están dentro de un do, repeat, etc) estos módulos serán representados como lo indica la figura anterior.



Un cuadro punteado nos ayudará a representar procesos en paralelo.



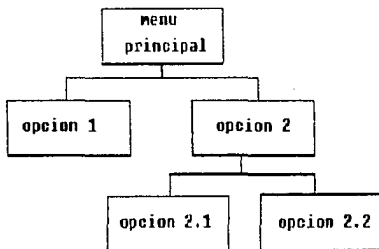
El símbolo (círculo sin relleno con flecha) representa datos o estructuras de datos enviadas de un módulo hacia otro, estos datos o estructuras deberán ser identificadas con el nombre que se les asigne en el punto de estructura de datos. El símbolo (círculo negro con flecha) nos indica el envío de variables de control.

Como ya mencionamos anteriormente esta es una metodología que nos ayudará a describir cada uno de los procesos que intervienen en nuestro diagrama de flujo de datos.

### 3.2.3 Descripción de menús

#### 3.2.3.1 Carta de estructura de menús

Debemos estructurar como irán relacionados, los diferentes menús que tendrá nuestro sistema; esto es, mediante un diagrama o bloques como el que se presenta.



**Nota :** Cada módulo es una pantalla diferente.

Se deberá definir como van relacionadas las pantallas de menú, con los procesos de nuestro diagramas de flujo de datos. Es recomendable que esta carta de estructura tenga como título el nombre del proceso que la genera.

Debemos definir una carta de estructura por cada proceso existente dentro de nuestro D.F.D.

En algunas cartas puede darse el caso de que los módulos pertenezcan a algún otro proceso es por eso que dentro de los recuadros interiores derechos indicaremos el proceso que lo genera, en caso de que este proceso que lo genera no sea el del título superior.

En cuanto a las entradas y salidas estas estén referidas al punto 4 del documento de análisis; esto es que aunque ahí fueron definidas a detalle en este punto pueden ser revisadas y más aún se deben especificar en que proceso están y en que momento aparece la pantalla de captura en el caso de entradas o la pantalla de reporte en el caso de salidas.

#### 3.2.3.2 Descripción de cada menú

Debemos referirnos a cada una de las pantallas que se declararon en el punto anterior, en caso de que la pantalla sea MENU se explicarán cuales son las opciones de

este menú y a que camino lleva cada una de estas opciones; en el caso de que la pantalla sea entrada, se definirá el orden de captura, tamaño de los campos, formatos y tipos de formatos, si los campos son obligatorios o no etc. además también es de gran ayuda nombrar cual es el destino de cada uno de los campos capturados; por último en el caso de ser salida la pantalla a la que nos referimos deberemos mencionar cual es la fuente de estos datos y como se debe de leer el dato.

#### 3.2.4 Sugerencias para el desarrollo de un buen "software"

- 1.- Que el diseño del sistema sea modular, considerando la independencia de módulos.
- 2.- Debe cumplir con los requerimientos solicitados.
- 3.- Debe ser eficiente en cuanto al manejo de los recursos requeridos.
- 4.- De fácil implementación, esto es que entre mayor sea la especificación de un módulo será más fácil implementar el sistema.
- 5.- Debe ser fácilmente modificable considerado como herramienta el diseño estructurado.

- 3.3 METODOLOGIA PARA LA IMPLEMENTACION DE SISTEMAS (6)**
- 3.3.1 Característica generales de la implementación**
  - 3.3.1.1 Elección del "software"**
  - 3.3.1.2 Tiempo de programación**
  - 3.3.1.3 Comentarios sobre la liberación**
- 3.3.2 Probando el "software"**
  - 3.3.2.1 Pasos en las pruebas del "software"**
    - 3.3.2.1.1 Pruebas de unidad**
    - 3.3.2.1.2 Pruebas de integración**
    - 3.3.2.1.3 Pruebas de validación**
    - 3.3.2.1.4 Probando las funciones del sistema**
  - 3.3.2.2 El arte de la depuración**
- 3.3.3 Mantenimiento del "software"**
  - 3.3.3.1 Mantenimiento mas fácil desde el desarrollo**
  - 3.3.3.2 Aspectos administrativos del mantenimiento del "software".**

**Definición.** Se entiende como implementación de los sistemas a la puesta en marcha de los mismos.

### 3.3.1 Características generales de la implementación

#### 3.3.1.1 Elección del "software"

Dentro de este trabajo, en el punto de principales componentes de software, tratamos el tema referente a las características del "software" que nos ofrece el equipo que vamos a utilizar.

Tomando en cuenta estas características, en la presente sección debemos decidir el "software" que mas nos conviene, la base de datos a utilizar y las utilerías que serán utilizadas en el desarrollo de nuestro sistema.

Debemos aclarar el porque de la elección, así como las especificaciones del "software" que optamos por utilizar, estas especificaciones se deben de plantear en forma detallada, para poder utilizar este documento como una guía rápida y completa del "software" con el que trabajaremos.

#### 3.3.1.2 Tiempo de programación

Para poder realizar este punto, nos basaremos en dos partes: la primera es tomar en cuenta el "pseudocódigo" realizado en el punto 3.1.5 titulado "Descripción Funcional", y la segunda es que en el punto anterior se definió el software a utilizar.

Tomando en cuenta estas dos partes, debemos dar aquí tiempos aproximados de desarrollo. Estos tiempos de desarrollo se deben dar por cada uno de los módulos que comprenden nuestro diagrama de flujo de datos, una medida adecuada para estos tiempos sería en semanas y de ser necesario, cada módulo se puede dividir en submódulos, para tener una mejor definición de los segmentos a realizar.

#### 3.3.1.3 Comentarios sobre la liberación

En esta parte se deberán aclarar todos los puntos especiales dentro de la liberación. Esto es, si por ejemplo; nuestro sistema es muy extenso y se puede utilizar por partes, sería conveniente especificar que primero se desarrollarán y se probarán las partes de uso mas frecuente en dado caso que se considere conveniente la liberación por partes.

Aquí también debemos definir tiempos y tipos de pruebas que se deben realizar al sistema para que pueda ser liberado. En el siguiente punto explicamos algunas pruebas que nosotros consideramos convenientes para revisar el software realizado.

### 3.3.2. Probando el "software"

La importancia de las pruebas del "software" no puede ser tomada a la ligera. Las pruebas del "software" son un elemento crítico para asegurar la calidad del mismo y representa la última revisión de especificaciones, diseño y codificación.

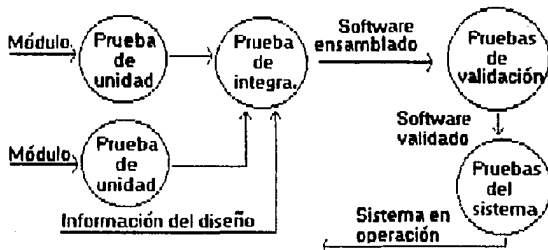
En este punto discutiremos dos partes que van muy relacionadas; la primera sería las pruebas del "software" las cuales serán divididas en pruebas de unidad, pruebas de integración, pruebas de validación y pruebas sobre las funciones del sistema; la segunda parte sería el arte de la depuración.

A continuación serán explicadas estas dos partes a detalle.

### 3.3.2.1 Pasos en las pruebas del "software"

Las pruebas del "software" nosotros las dividimos para mayor claridad en cuatro pasos que son implementados secuencialmente. Inicialmente, la primera prueba enfoca a cada módulo en forma individual, conocemos como módulo a un conjunto de instrucciones agrupadas con el fin de realizar un trabajo específico; a esta prueba de los módulos en forma individual, le llamamos prueba de unidad. Lo siguiente, es que los módulos sean integrados o ensamblados para formar un paquete de software probando aquí el acoplamiento, a esta prueba se le conoce como prueba de integración. La siguiente prueba a realizar sería la de la validación asegurando con esto que el software funcione completamente y ejecute los requerimientos. La última prueba sería la prueba del sistema, la cual verifica que todos los elementos de la mailla realicen su función perfectamente.

La siguiente gráfica muestra lo explicado anteriormente:



A continuación son explicadas cada una de las cuatro pruebas.

#### 3.3.2.1.1 Pruebas de unidad

Las pruebas de unidad realizan su verificación en la parte más pequeña del diseño del "software", el módulo. Durante las pruebas de unidad, tres características primarias deben ser evaluadas.

- La estructura de datos local
- Las condiciones límite
- Las interfases del módulo

Ahora, cuando un módulo ejecuta procesos de entrada/salida en archivos, el autor Myers (7) en su libro titulado "The art of software testing" plantea las siguientes preguntas para verificar la unidad del módulo.

- ¿Son correctos los atributos del archivo?
- ¿Esta correctamente escrita la sentencia "OPEN"?
- ¿Se abrieron los archivos antes de usarse?
- ¿Se maneja la condición de fin de archivo?
- ¿Se manejan los errores de entrada/salida?

La estructura local de los datos de un módulo comúnmente es fuente de errores. Los errores más comunes se pueden incluir en alguna de las siguientes categorías.

- Declaración inconsistente
- Valores asignados incorrectos
- Nombre de variable incorrecto
- Tipo de datos inconsistente.

#### 3.3.2.1.2 Pruebas de integración.

Se requiere probar el flujo de datos a través de los módulos antes de comenzar otra prueba, si el flujo de datos no es correcto no se deben realizar las otras pruebas. El autor Myers (7) propone las siguientes preguntas para probar el flujo de datos entre módulos.

- ¿El número de parámetros de entrada del módulo llamado es igual al número de parámetros de salida del módulo que llama?
- ¿Los parámetros de entrada y los de salida son del mismo tipo?
- ¿El orden de los parámetros es el mismo en ambos módulos?
- ¿Son respetadas las variables globales dentro del módulo o existen conflictos?

#### 3.3.2.1.3 Pruebas de validación

Al terminar las pruebas de integración, el "software" se encuentra completamente ensamblado como un solo paquete; los errores de interfase fueron descubiertos y corregidos, por lo tanto podemos comenzar con las pruebas de validación.

Las pruebas de validación toman al "software" como si fuera una caja negra e intentan demostrar que cumple con los requerimientos del usuario, no planteamos una metodología puesto que se trata simplemente de probar cada módulo y verificar si cumple con los requerimientos planteados, una vez terminada la prueba de cada módulo se pueden tomar dos acciones, si resultado satisfactoria la prueba se continúa probando, en caso de no satisfacer las necesidades se realizará una lista de problemas.



### 3.3.2.1.4 Probando las funciones del sistema

Algunos pasos que se recomiendan para probar las funciones del sistema son:

- Crear una serie de datos malos para ver que clase de problemas causan
- Grabar los datos y los resultados de las pruebas para tenerlos como evidencia.
- Participar en la planeación y diseño de las pruebas para asegurarnos que estas son las adecuadas.

Por último al terminar estas pruebas realizadas por algunas personas incluyendo a la persona que desarrollo el "software" y el usuario final, se debe dejar el sistema en manos del usuario final para que él lo maneje.

### 3.3.2.2 El arte de la depuración

Las pruebas del "software" son un proceso que puede ser planeado sistemáticamente, además estas pruebas son diseñadas para descubrir errores. La depuración ocurre como consecuencia de los resultados de las pruebas.

La depuración tiene como objetivo encontrar y corregir las causas de los errores del "software", el objetivo es alcanzado por una serie de evaluaciones sistemáticas, intuición y suerte. En general, tres tipos de depuración pueden ser propuestas:

- La fuerza bruta
- Eliminación de causas
- "Backtracking" (regresar por lo ya andado)

La fuerza bruta es generalmente el método más común, esto se manifiesta en que si se encuentra algún tipo de error, este es atacado llenando el programa de sentencias "write" hasta tener encerrado al problema.

La segunda más usual es la eliminación de causas esta se manifiesta por la inducción o la deducción, se hace una lista de las posibles causas y se van eliminando, conforme se realizan pruebas a cada una de las posibles causas.

El "backtracking" es el menos usual, se comienza por colocarse en donde empieza el programa y se sigue el código fuente hasta encontrar el error.

### 3.3.3 Mantenimiento del "software"

El término "mantenimiento del software" es usado para describir las actividades que ocurren después de entregar el "software" al cliente. Generalmente el desarrollo del "software" lleva alrededor de 1 a 2 años y por otro lado el mantenimiento se da por un espacio de 5 a 10 años.

Las actividades del mantenimiento involucran hacer cambios al "software", adaptaciones para nuevos ambientes y correcciones a problemas.

#### 3.3.3.1 Mantenimiento más fácil desde el desarrollo

Algunas actividades ejecutadas desde el desarrollo del "software" hacen más fácil el mantenimiento. Algunas de estas actividades son enlistadas a continuación.

**Actividades en el análisis**

- Desarrollo estándar
- Estimar costos de mantenimiento
- Especificar calidad de los procedimientos

**Actividades del diseño**

- Enfatizar claridad y modularidad
- Diseño fácil
- Uso estándar de notación para documentos, flujos de datos, funciones, estructuras e interconexiones.

**Actividades de la implementación**

- Uso de entradas y salidas únicas
- Uso simple y claro del código
- Uso simbólico de constantes
- Documentación interna de los listados

**Otras actividades**

- Desarrollo de una guía de mantenimiento
- Desarrollo de una prueba

### 3.3.3.2 Aspectos administrativos del mantenimiento del "software"

Dentro del mantenimiento del software existen algunos aspectos administrativos, los cuales serán tratados en este punto. Las actividades del mantenimiento en el "software" generalmente ocurren como respuestas a cambios. Los requerimientos de cambio pueden ser descritos por el siguiente algoritmo:

- iniciar los requerimientos de cambio
- analizar los requerimientos
- Si requerimiento no válido ENTONCES
  - requerimiento cerrado
- DE OTRA MANERA
  - requerimientos y recomendación se programan
  - ejecución de requerimientos
- Si cambios hechos se aprueban ENTONCES
  - actualizar archivo maestro
  - extender documento actualizado
- DE OTRA MANERA
  - se vuelven a programar los cambios
- FIN

Los requerimientos de cambios son generalmente iniciados por los usuarios. Un requerimiento de cambio puede solicitar mejoras, adaptaciones o corrección de errores. Mejoras y adaptaciones en gran cantidad puede requerir un nuevo análisis y nuevas negociaciones con el cliente.

Dentro del aspecto administrativo manejaremos una tarjeta de control de cambios, en ella registraremos todos los requerimientos de cambios, en la tarjeta pondremos los

canbios denegados y los cambios aprobados. Las partes que forman la tarjeta de control de cambios, dependen del software, la estructura organizacional y la gente involucrada.

También debemos manejar un sumario de los requerimientos de cambios, reportaremos en el las emergencias; incluiremos fechas y descripción de requerimientos.

El último punto que manejaremos será la organización de los programadores de mantenimiento, estos programadores es recomendable que sean las personas que conocen los módulos y el sistema en general, esto ahorrará tiempo y esfuerzos.

## **4. DESARROLLO DEL SISTEMA ADMINISTRACION COSTO DE OBRA**

#### **4.1 ANALISIS DEL SISTEMA DE ADMINISTRACION COSTO DE OBRA**

Como lo indica la metodología para el Análisis de Sistemas este capítulo contiene los siguientes puntos:

- 4.1.1 Antecedentes del Sistema de Administración Costo de Obra**
- 4.1.2 Objetivos y alcances del Sistema de Administración Costo de Obra**
- 4.1.3 Descripción general del Sistema de Administración Costo de Obra**
- 4.1.4 Descripción de entradas/salidas del Sistema de Admon Costo de Obra**
- 4.1.5 Descripción funcional del Sistema de Administración Costo de Obra**
- 4.1.6 Análisis de la información del Sistema de Administración Costo de Obra**
- 4.1.7 Diccionario de datos del Sistema de Administración Costo de Obra**
  - 4.1.7.1 Estructuras de datos del Sistema de Administración Costo de Obra**
  - 4.1.7.2 Flujos de datos del Sistema de Administración Costo de Obra**
- 4.1.8 Recursos y calendario para desarrollar el Sistema de Admon Costo de Obra**
- 4.1.9 Anexos del Sistema de Administración Costo de Obra**

#### 4.1.1 Antecedentes del Sistema de Administración Costo de Obra

Una de las actividades de la Dirección General de Obras y Servicios Generales (D.G.O.S.G.), consiste en llevar a cabo la administración de las obras civiles que ahí se realizan.

El volumen de información involucrada es considerable, lo cual trae como consecuencia que el proceso administrativo sea lento y muy laborioso.

Como una alternativa de solución ha esta problemática se ha propuesto el desarrollo de un sistema que automatice todo este proceso para un buen desarrollo de la administración.

Podemos resumir el desarrollo de este proceso administrativo en las siguientes etapas:

1. Cuando se va a realizar una obra, se genera una convocatoria-concurso con las características de la obra a realizar.
2. La D.G.O.S.G. genera sus propios cálculos de costos mediante el llenado de la forma U-7.
3. Una vez que se eligió el contratista que realizará la obra, se plantea su duración mediante una ruta crítica.
4. Para la realización de las obras se necesita tener un valor estimado de los precios unitarios que intervienen en la obra y un comparativo del estimado precio D.G.O.S.G. y el precio del contratista.
5. Es necesario también contar con una cantidad aproximada de los insumos que formarán parte de la obra, término conocido como explosión de insumos.
6. Existe una gran variedad de información que debe ser distribuida por varios departamentos, como lo son precios unitarios, insumos, conceptos, explosión de insumos, etc.
7. Conforme la obra avanza, la ruta crítica original nos va mostrando si el avance es el programado o no lo es, si el avance no es el estimado la ruta crítica debe ser modificada en algunas ocasiones.

#### 4.1.2 Objetivos y alcances del sistema del Sistema de Administración Costo de Obra

##### Objetivos

El principal objetivo del sistema consiste en proporcionar una herramienta de gran uso a quienes se encuentran involucrados en la administración de la obra civil. Por otro lado se pretende evitar el desperdicio de recursos humanos en cuestiones como, cálculo de precios unitarios y el cálculo-gráfico de la ruta crítica, ya que esto implica costos para la D.G.O.S.G. Además se pretende que la información que sea necesaria para la toma de decisiones se tenga en el lugar y momento adecuado.

Logrando lo anterior se evitará que se tomen decisiones equivocadas por falta de información, además se tendrá toda la información en un solo banco de datos al cual podrán recurrir todos los interesados en el momento que lo requieran, este punto es muy importante porque actualmente el manejo de información es aislado para cada subdirección y no se realiza una comparación de precios ni de proveedores como debería de ser para disminuir costos.

Refiriéndose a los objetivos particulares, se espera que el sistema contemple los siguientes puntos en forma de reportes:

- Cálculo de precios unitarios
- Cálculo de los costos directos
- Explosión de insumos requeridos en la obra
- Cálculo de la ruta crítica
- Cálculo del costo total de la obra
- Cálculo del costo parcial de la obra (por partidas)
- Algunos puntos complementarios

#### Alcances.

Con este sistema se pretende tener un mayor control sobre las obras que se realizan en la D.G.O.S.G., también se pretende dar solución a algunos problemas que se presentan con el manejo de la información y con el cálculo de los precios unitarios de cada subdirección, se espera que los procesos administrativos relacionados con las obras sean más eficientes.

Al mecanizar mediante la computadora algunos cálculos, los recursos humanos se verán un poco más holgados para la realización de algunas otras actividades.

La información relacionada con la administración de las obras se tendrá al alcance de quién la solicite y en forma casi instantánea.

#### 4.1.3 Descripción General del Sistema de Administración Costo de Obra

Nuestro sistema para su mejor manejo estará presentado al usuario en forma de árbol, este árbol lo desglosamos a continuación para que el usuario comience a conocer los puntos que lo componen. Estará desarrollado de la siguiente manera.

Nuestro primer menú o menú principal tendrá la posibilidad de elegir alguna de las siguientes opciones:

1. Manejo de proyectos
2. Control de proveedores
3. Catálogos maestros
4. Catálogos particulares
5. Reportes
6. Ruta crítica

1. Manejo de proyectos. Maneja las características generales de una obra como son: fecha de inicio, descripción, fecha de creación de archivos, dependencia para la que se realiza, etc., se tienen las siguientes opciones:

- 1.1. Altas. Cuando se necesita comenzar una nueva obra, se recurrirá a este módulo, aparecerá un formato el cual nos pedirá la información general de la obra.
- 1.2. Bajas. Con dar la clave de la obra esta podrá ser eliminada, como es muy delicado eliminar obras completas, contaremos con un implemento de seguridad para proteger las obras.
- 1.3. Modificaciones. En caso de que ocurra algún error en la captura o se necesite algún cambio de la información, podremos utilizar este módulo.

1.4. Consultas. Este módulo es sumamente útil, puesto que nos permitirá verificar la información, tener la información a la mano para la toma de decisiones, también nos permitirá tener bastante información en cuestión de segundos en caso de solo necesitar una simple consulta.

2.- Control de proveedores. Manejará toda la información referente a los proveedores y contratistas con los que trabaja la D.G.O.S.G., esta información comprende, clave del proveedor, productos que suministra, dirección, persona a contactar, etc., las opciones son:

- 2.1. Alta. Si es necesario introducir un nuevo proveedor en nuestro archivo recurrimos a este módulo, accedendo toda la información del proveedor.
- 2.2. Bajas. Tecleando la clave del proveedor que queremos eliminar de nuestro archivo este se eliminará.
- 2.3. Modificaciones. En caso de querer cambiar algún dato de un proveedor utilizaremos este punto, lo único que no se puede cambiar es la clave del proveedor y el nombre.
- 2.4. Consultas. Para conocer las características del proveedor, nombre, dirección, etc. utilizaremos este módulo.

Existirán dos tipos de catálogos, los primeros se denominarán catálogos maestros y existirá uno para insumos y uno para conceptos. Los segundos serán los catálogos particulares y serán referidos a cada obra, nuestros catálogos particulares serán de insumos y conceptos, aquí se incluye al manejo del archivo de presupuesto por ser también un presupuesto por obra.

3.- Catálogos maestros. Maneja toda la información de los catálogos maestros de insumos y de conceptos, así como el precio que registró en el catálogo de conceptos y las actualizaciones de precios, se manejan las siguientes opciones:

- 3.1. Insumos. El catálogo maestro de insumos es manejado en este punto, para su manejo se deberán desarrollar los correspondientes módulos de altas, bajas, cambios y consultas.
- 3.2. Conceptos. Los conceptos maestros se podrán manejar en este punto, para hacer este manejo se deben de tener los siguientes módulos: altas, bajas, cambios y consultas.
- 3.3. Precio que rige el catálogo. El sistema maneja cuatro diferentes precios de insumos, un precio por cada subdirección (Conservación, Construcción y Obras externas) y un último precio denominado precio D.G.O., siendo este el precio que rige para cualquier cálculo posterior, en caso que se necesite que el precio de alguna subdirección domine en los futuros cálculos se recurre a este módulo, las opciones deberán ser por tanto:

- 3.3.1. Conservación
- 3.3.2. Obras externas
- 3.3.3. Construcción

Siendo que al elegir alguna opción, los precios de esta subdirección serán copiados al espacio designado para precios D.G.O.



- 3.4. Actualizaciones de precios. Dentro de nuestro catálogo maestro de conceptos, este precio puede ser afectado por un porcentaje de indirectos y por un porcentaje de I. V. A. que pueden variar, estos dos porcentajes se actualizarán aquí y automáticamente se recalcularán los precios de los conceptos afectados por los nuevos porcentajes.

4. Catálogos particulares. Estos catálogos particulares son los que pertenecen a cada una de las obras que se dieron de alta en el punto uno "Catálogo costo de obra" y se tendrán para su manejo las siguientes opciones:

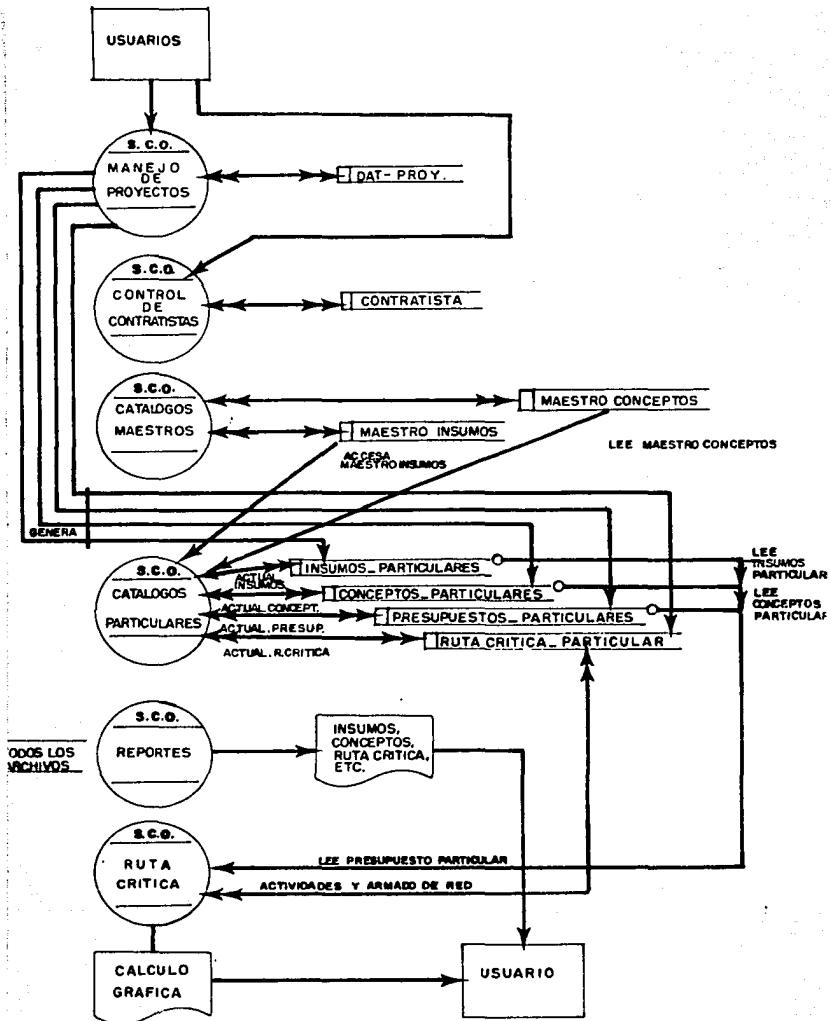
- 4.1. Presupuesto. Dentro de una obra en particular, uno de los puntos que nos interesa tener controlado es el presupuesto de la misma, para llevar a cabo este control necesitaremos los siguientes módulos:
- 4.1.1. Altas. Cuando nuestro presupuesto es nuevo o si queremos meter nuevos conceptos recurriremos a este punto.
- 4.1.2. Bajas. En caso de querer eliminar conceptos que ya forman parte de nuestro presupuesto utilizaremos este módulo.
- 4.1.3. Cambios. Si alguno de los conceptos que forman nuestro presupuesto debe ser cambiado en cantidad, nombre, etc., se recurrirá a este módulo.
- 4.1.4. Consultas. Módulo usado para consultar los conceptos que forman nuestro presupuesto, el subtotal por partida, el total del presupuesto, etc.
- 4.2. Conceptos. Aquí podremos manejar todo lo relacionado con los conceptos particulares de una obra, no se podrá dar de alta un concepto porque las altas se realizan sólo cuando damos de alta los conceptos en el presupuesto, pero si podemos realizar bajas, cambios y consultas, para hacer esto se deberán desarrollar los módulos correspondientes.
- 4.3. Insumos. Los insumos particulares podrán ser modificados y consultados en este punto, por seguridad no se podrán dar de baja y se podrán dar de alta al mismo tiempo que se realiza el alta de los conceptos en el presupuesto.
- 4.4. Actualizaciones de precios. Este punto llevará a cabo las mismas acciones que el punto 3.4. solamente que aquí afectaremos un catálogo de conceptos particular y en el 3.4. se afectará el catálogo de conceptos maestro.
- 5.- Reportes. Todos los reportes que se soliciten, siempre y cuando sean viables, se incluirán aquí. Podremos mencionar los más sobresalientes. Reporte de insumos, reporte de conceptos, reporte de presupuesto, explosión de insumos, etc.
- 6.- Ruta crítica. En caso de necesitar el cálculo y la gráfica de la ruta crítica de una obra en particular, entraremos a este módulo, el cual nos presentará las siguientes opciones:
- 6.1. Altas de actividades. Aquí indicaremos cuáles de las actividades que forman parte del presupuesto particular, integrarán nuestra ruta crítica. Para facilidad en el manejo de la ruta ligaremos la clave de la actividad de 9 dígitos con una clave del tipo letra-letra de dos dígitos.
- 6.2. Armado de la red. La ruta crítica se obtiene de una red de actividades, para poder armar esta red usaremos este punto, señalando la actividad que necesitaremos y las actividades anteriores y posteriores a esta.
- 6.3. Cálculo y gráfica. Una vez armada la red de actividades y obtenidos los tiempos de duración de las actividades del catálogo de conceptos, se puede realizar el

cálculo de la ruta crítica, con el cálculo hecho sólo nos faltaría graficar, estos puntos se obtendrán aquí.

6.4. Cambios de tiempos. Si los tiempos de duración de las actividades son muy grandes estos se pueden disminuir aumentando cuadrillas, esta operación se realiza en este punto.

6.5. Bajas de conceptos. En caso de que algún concepto esté mal dado o ya no se realice, se puede recurrir a este módulo. Como es muy delicado en este caso el manejo de las bajas, se requerirá de una clave de acceso.

Lo anterior sería la narrativa sencilla, requerida en la descripción general, a continuación presentaremos el diagrama de flujo de datos.



Cabe mencionar que lo anterior es una vista global del sistema, y que el tema que nosotros nos propusimos desarrollar en el presente trabajo es el de Administración costo de obra, el cual estaría formado por la ruta crítica y la explosión de los insumos.

#### 4.1.4 Descripción de entradas/salidas del Sistema de Administración Costo de Obra

Para realizar la descripción de entradas salidas, nos basaremos en la teoría expuesta con el mismo nombre dentro del capítulo de metodología del análisis, usaremos las formas definidas ahí y las especificaciones que en dicho punto se plantearon.

Dentro de la administración costo de obra, utilizaremos las siguientes formas de entrada/salida.

Librería: formas.lib      Forma: menú.form

Campos:

<u>Nombre del campo</u>	<u>Leyenda</u>	<u>Tamaño</u>
campo1	manejo de proyectos [1] control de proveedores [2] catálogos maestros [3] catálogos particulares [4] reportes [5] ruta crítica [6]	1

La pantalla anterior se utilizará para poder elegir alguna de las opciones que nos muestra dependiendo de lo que nosotros necesitemos hacer.

Librería: formas.lib      Forma: ella\_ruta\_1.form

Campos:

<u>Nombre del Campo</u>	<u>Leyenda</u>	<u>Tamaño</u>
campo 1	Clave del concepto	6
campo 2	Clave de la actividad	2
campo 3	Descripción	50
campo 4	Unidad	3
	Descripción de O. de mano	3 renglones
campo 5	Tiempo por unidad (horas)	3
campo 6	Tiempo por unidad (minutos)	2
campo 7	Volumen por jornada	6
campo 8	Duración en jornadas	6
campo 9	Número de cuadrillas	2

La pantalla ellas\_ruta\_1.form se utilizará para calcular los conceptos que forman parte de la ruta crítica, desplegar las duraciones de estos conceptos y asignarles una clave de dos dígitos que no proporcione ninguna complicación para el usuario.

Librería: Formas.lib      Forma: Alta\_ruta\_2.form

Campos:

<u>Nombre del Campo</u>	<u>Leyenda</u>	<u>Tamaño</u>
Campo1	Actividad 1	2
Campo2	Actividad 2	2
Campo3	Actividad 3	2
Campo4	Actividad 4	2
Campo5	Actividad 5	2
Campo6	Descripción 1	50
Campo7	Descripción 2	50
Campo8	Descripción 3	50
Campo9	Descripción 4	50
CampoA	Descripción 5	50
CampoB	Duración 1	5
CampoC	Duración 2	5
CampoD	Duración 3	5
CampoE	Duración 4	5
CampoF	Duración 5	5
CampoG	Posterior 1,1	2
CampoH	Posterior 1,2	2
.	.	.
.	.	.
.	.	.
CampAZ	Posterior 5,5	2

Esta pantalla nos servirá para armar la red de actividades que forman nuestra obra, como ayuda auxiliar se mostrará el nombre del concepto y la duración que elegimos darle.

Librería: Formas.lib      Forma: Cambios\_ruta.form

Campos:

<u>Nombre del Campo</u>	<u>Leyenda</u>	<u>Tamaño</u>
Campo 1	Clave del concepto	8
Campo 2	Clave de la actividad	2
Campo 3	Descripción	50
Campo 4	Unidad	3
	Descripción de O. de mano	3 renglones
Campo 5	Tiempo por unidad (horas)	3
Campo 6	Tiempo por unidad (minutos)	2
Campo 7	Volumen por jornada	5
Campo 8	Duración en jornadas	5
Campo 9	Número de cuadrillas	2

En caso de que el resultado entregado como ruta crítica no sea el esperado (en duración total) se podrá cambiar la duración de las actividades que forman la obra, cambiando el número de cuadrillas en la pantalla anterior.

Librería: Formos.lib      Forma: Bajas\_ruta.form

Campos:

<u>Nombre del Campo</u>	<u>Leyenda</u>	<u>Tamaño</u>
Campo 1	Clave de acceso para eliminar	6
Campo 2	Concepto a eliminar	2

Si necesitamos eliminar algún concepto dentro de nuestra obra, utilizaremos la pantalla anterior. Como al eliminar un concepto la red puede sufrir alteraciones no deseables debemos ser precavidos, es por esto que utilizaremos una clave de acceso.

Librería: Formas.lib      Forma: Menú\_ruta.form

Campos:

<u>Nombre del Campo</u>	<u>Leyenda</u>	<u>Tamaño</u>
Campo 1	Altas 1	1
	Altas 2	2
	Bajas	3
	Cambios	4
	Proceso y gráfica	5

Este menú nos permitirá elegir una de las siguientes opciones:

- altas 1
- altas 2
- bajas actividad
- cambios de dirección
- realiza ruta y gráfica

Librería: formas.lib      forma: menú\_reportes

Campos:

<u>Nombre del campo</u>	<u>Leyenda</u>	<u>Tamaño</u>
campo1	catálogos maestros	[1] 1
	catálogos particulares	[2]
	explosión de insumos	[3]

Aquí podemos elegir algún reporte en particular, ya sea insumos, conceptos, etc., en este caso el que nos interesa es un reporte llamado - explosión de insumos -.

En cuanto al manejo de las salidas tendremos las siguientes:

La primer salida a especificar será la de explosión de insumos.

sistema: control costo de obra

títulos  
fecha: "explosión de insumos" página #: \_\_\_\_\_  
clave de la obra a reportar: \_\_\_\_\_  
descripción de la obra: \_\_\_\_\_

clave de insumo	reporte a precio unitario o a costo directo descripción del insumo	unidad	cantidad usada en obra
_____	_____	_____	_____
_____	_____	_____	_____

dispositivo: impresora periodicidad: uno cada obra  
volumen : de 1 a 10 hojas tipo de proceso: en línea  
tiempo de resp.: inmediato proceso generador: ruta crítica

La siguiente salida a especificar es la de la ruta crítica, lo dividiremos en dos partes; la primera sería la entrega de cálculos.

sistema: control costo de obra

títulos  
clave de la obra: \_\_\_\_\_

# act	cve act	actividades post1...post5	duración	ip	tp	ir	tr	holgura
_____	_____	_____	_____	_____	_____	_____	_____	_____
_____	_____	_____	_____	_____	_____	_____	_____	_____

dispositivo: impresora periodicidad: 3 máximo por obra  
volumen : 1 a 3 hojas tipo de proceso: en línea  
tiempo de resp.: inmediato proceso generador: ruta crítica

en la segunda parte se realiza la gráfica

-----  
sistema : control costo de obra  
-----

mes==>  
semana de...hasta==> 13    18 20    25 27    3...  
no de semana==> 1        2        3...

act. descripción

-----  
-----  
-----  
-----

Zona de la gráfica.

-----  
dispositivo : impresora  
volumen : de 1 a 10 hojas  
tiempo de resp. : inmediato

periodicidad : 3 máximo cada obra  
tipo de proceso : en línea  
proceso generador : ruta crítica  
-----

#### 4.1.5 Descripción funcional del Sistema de Administración Costo de Obra

La descripción funcional, fué explicada en teoría dentro del capítulo titulado "Metodología para el Análisis de Sistemas", basándonos en esta teoría, realizamos el siguiente trabajo.

Descripción funcional de la parte de explosión de insumos.

- 1.0 asigna salida a impresora
- 1.1 lee de que obra vamos a realizar la explosión de insumos
- 1.2 REALIZA REPORTE
  - 1.2.1 inicializa variables
  - 1.2.2 escribe fecha
  - 1.2.3 ABRE ARCHIVOS
    - 1.2.3.1 abre insumos generales
    - 1.2.3.2 abre datos de los proyectos
    - 1.2.3.3 abre conceptos particulares
    - 1.2.3.4 abre presupuesto particular
  - 1.2.4 IMPRIME ENCABEZADO
    - 1.2.4.1 escribe títulos
    - 1.2.4.2 lee fecha del sistema e imprimela
    - 1.2.4.3 escribe "EXPLOSION DE INSUMOS"
    - 1.2.4.4 lee datos de la obra de archivo
    - 1.2.4.5 si tipo de reporte = 1 entonces
      - 1.2.4.5.1 escribe "REPORTE A COSTO DIRECTO"
    - 1.2.4.6 de otra manera
      - 1.2.4.6.1 escribe "REPORTE A PRECIO UNITARIO"
  - 1.2.5 repite



- 1.2.5.1 lee registro de presupuesto
  - 1.2.5.2 ACTUALIZA ARREGLO
    - 1.2.5.2.1 lee conceptos con llave tomada del registro de presupuesto
    - 1.2.5.2.2 desgloza insumos
    - 1.2.5.2.3 si insumo es básico entonces
      - 1.2.5.2.3.1 guarda en tabla anexa
      - 1.2.5.2.4 de otra manera
        - 1.2.5.2.4.1 busca en arreglo y actualiza cantidad  $\text{cantidad} = \text{cantidad} + (\text{volumen} * \text{rendimiento})$
  - 1.2.6 hasta fin de arreglo auxiliar
  - 1.2.7 repite
    - 1.2.7.1 ACTUALIZA ARREGLO
    - 1.2.7.2 concepto = insumos\_básico
    - 1.2.7.3 desgloza insumos
    - 1.2.7.4 si insumos es básico entonces
      - 1.2.7.4.1 guarda al final de la tabla anexa
      - 1.2.7.5 de otra manera
        - 1.2.7.5.1 busca en arreglo y actualiza cantidad
  - 1.2.8 hasta fin de arreglo auxiliar
  - 1.2.9 imprime insumos con cantidad, unidad y descripción
  - 1.2.10 CIERRA ARCHIVOS
    - 1.2.10.1 cierra insumos generales
    - 1.2.10.2 cierra datos de los proyectos
    - 1.2.10.4 cierra conceptos particulares
    - 1.2.10.5 cierra presupuesto particular
- 1.3 fin de la explosión de insumos

#### Descripción funcional de la parte de ruta crítica

Para referirnos a la ruta crítica, la dividiremos en las siguientes partes :

- altas de actividades a la ruta crítica
- armado de la red
- bajas de actividades
- cambios en las duraciones
- proceso ruta crítica
- grafica ruta crítica

primero daremos la descripción funcional de altas de actividades.

- 1.0 Inicializa variables
- 2.0 Abre forma de captura y despliega
- 3.0 lee clave del concepto de pantalla si clave = tecla fin entonces terminamos
- 4.0 lee del archivo de conceptos y presupuesto con la llave anterior la información
- 5.0 valida si el concepto se dió de alta en el presupuesto
- 6.0 si el concepto no está en el presupuesto entonces
  - 6.1 error "ese concepto se debe dar de alta primero en el presupuesto"
  - 7.0 de otra manera
    - 7.1 despliega información del concepto en pantalla
    - 7.2 asigna clave de dos letras por facilidad y despliega
    - 7.3 con información del archivo de presupuesto calcula duración y despliega
    - 7.4 despliega número de cuadrillas igual a uno
    - 8.0 lee teclas
    - 9.0 Repite
      - 9.1 Case (teclas)
        - 9.1.1 Cncl: no guardamos nada sólo limpiamos pantalla

- 9.1.2 Sigue grabamos información en archivo de administración y limpiamos pantalla
- 9.1.3 1..99:nos indica que este es el número de cuadrillas a utilizar, recalculamos duración y desplegamos
- 10.0 hasta que (tecla = Cncl) ó (tecla = Sig) )
- 11.0 fin de ellas de actividades

#### Descripción funcional del armado de la red.

- 1.0 Abre archivos
- 2.0 Inicializa variables
- 3.0 Abre forma de captura y despliega
- 4.0 Repite
  - 4.1 lee secuencialmente las actividades del archivo de administración
- 4.2 despliega información
- 4.3 lee campos de captura
- 4.4 graba nueva información en archivo de administración
- 5.0 hasta que tecla sea igual a fin
- 6.0 Cierra archivos
- 7.0 fin de armado de la red

#### Descripción funcional de bajas de actividades

- 1.0 Abre archivos
- 2.0 Inicializa variables
- 3.0 Abre forma de captura y despliega
- 4.0 lee clave de la actividad a eliminar
- 5.0 pregunta "¿estas seguro de querer eliminar?"
- 6.0 si respuesta = no entonces
  - 6.1 terminamos
- 7.0 de otra manera
  - 7.1 pregunta "¿ clave de acceso para eliminación ?"
- 7.2 si clave correcta entonces
  - 7.2.1 eliminamos del archivo
- 7.3 de otra manera
  - 7.3.1 desplegamos "error tu no puedes eliminar el concepto"
- 8.0 Cierra archivos
- 9.0 fin de bajas de actividades

#### Descripción funcional de cambios de duración

- 1.0 Inicializa variables
- 2.0 Abre archivos
- 3.0 Abre y despliega forma "alla a ruta critica"
- 4.0 lee la actividad a hacerle cambios
- 5.0 obtien información de esta actividad del archivo
- 6.0 si error en lectura entonces
  - 6.1 "ese concepto no existe" y termina proceso
- 7.0 de otra manera
  - 7.1 continúa proceso normal
- 8.0 despliega información de la actividad
- 9.0 repite
  - 9.1 lee nuevo número de cuadrillas
  - 9.2 calcula y despliega nueva duración
- 10.0 hasta que tecla sea igual a return
- 11.0 cierra archivos

## 12.0 fin de la rutina cambios

### Descripción funcional de proceso ruta crítica

- 1.0 inicializa variables
- 2.0 abre archivos
- 3.0 repite
  - 3.1 lee secuencialmente archivo de ruta crítica
  - 3.2 asigna en un arreglo actividades posteriores, duración y concepto
- 4.0 hasta que sea fin de archivo
- 5.0 inicialización de ajustes
- 6.0 llenamos el renglón de ajustes basados en las actividades posteriores
- 7.0 buscamos que actividad tiene ajuste cero y nos posicionamos ahí
- 8.0 buscamos si alguna otra actividad tiene ajuste cero
- 9.0 si dos o mas actividades tienen ajuste cero entonces
  - 9.1 escribe "error, la ruta no tiene un inicio lógico"
  - 9.2 termina proceso
- 10.0 de otra manera
- 10.1 continúa proceso normal
- 11.0 REORDEN
  - 11.1 repite
    - 11.1.1 actividad = sig. actividad según el reorden
    - 11.1.2 repite
      - 11.1.2.1 tomamos la actividad posterior
      - 11.1.2.2 nos posicionamos en la actividad posterior
      - 11.1.2.3 restamos uno al ajuste
      - 11.1.2.4 si ajuste = 0 entonces
        - 11.1.2.4.1 asignamos siguiente número de reorden a esta actividad
      - 11.1.3 hasta que terminemos con las actividades posteriores
    - 11.2 hasta que terminemos con todas las actividades de la red
- 12.0 fin del proceso reorden
- 13.0 CALCULA TIEMPOS PROXIMOS
  - 13.1 repite i=1 hasta número total de actividades
    - 13.1.1  $\text{arreglo}[i].\text{terminación\_próxima} = \text{arreglo}[i].\text{iniciación\_próxima} + \text{arreglo}[i].\text{duración}$
    - 13.1.2 si  $\text{arreglo}[i+1].\text{iniciación\_próxima} < \text{arreglo}[i].\text{terminación\_próxima}$  entonces
      - 13.1.2.1  $\text{arreglo}[i+1].\text{iniciación\_próxima} = \text{arreglo}[i].\text{terminación\_próxima}$
  - 13.2 fin de repite
- 14.0 fin de calcula tiempos próximos
- 15.0 CALCULA TIEMPOS REMOTOS
  - 15.1 repite i=número de actividades decrementando hasta 1
    - 15.1.1  $\text{arreglo}[i].\text{terminación\_remota} = \text{arreglo}[i].\text{terminación\_próxima}$
    - 15.1.2 si  $\text{arreglo}[i].\text{terminación\_remota} > \text{arreglo}[i-1].\text{iniciación\_remota}$  entonces
      - 15.1.2.1  $\text{arreglo}[i].\text{terminación\_remota} = \text{arreglo}[i-1].\text{iniciación\_remota}$
    - 15.1.3 fin del si
  - 15.2 fin del repite
- 16.0 fin de calcula tiempos remotos
- 17.0 CALCULA HOLGURAS
  - 17.1 repite i=1 hasta número de actividades
    - 17.1.1  $\text{arreglo}[i].\text{holgura} = \text{arreglo}[i].\text{terminación\_remota} - \text{arreglo}[i].\text{terminación\_próxima}$
  - 17.2 fin de repite
- 18.0 fin de calcula holguras
- 19.0 fin de proceso ruta crítica

### Descripción funcional de realiza gráfica

- 1.0 inicializa variables

- 2.0 abre archivos
- 3.0 abre y despliega forma de captura
- 4.0 lee día de semana y fecha en que debe empezar la gráfica
- 5.0 salta al principio de la próxima hoja
- 6.0 imprime encabezados
- 7.0 repite
  - 7.1 repite i=1 hasta número de actividades
    - 7.1.1 si ya imprimimos 50 actividades entonces
      - 7.1.1.1 saltamos hoja
    - 7.1.2 fin del si
    - 7.1.3 lee de archivo la información de la siguiente act.
    - 7.1.4 redondea la duración
    - 7.1.5 calcula blancos necesarios
    - 7.1.6 decide tipo de símbolo a imprimir /, \, \*, <, >.
    - 7.1.7 calcula número de símbolos a imprimir
    - 7.1.8 imprime
  - 7.2 fin del repite
  - 7.3 terminamos la gráfica sí o no
- 8.0 hasta que grafica este completa
- 9.0 fin del proceso realiza gráfica

#### 4.1.6 Análisis de la información del Sistema de Administración Costo de Obra

Primero se debe definir toda la información que se necesita para el desarrollo de nuestro sistema.

Para el manejo de los presupuestos particulares necesitamos:

- clave del concepto que formará parte del presupuesto
- número de partida a la que se asignará este concepto
- volumen que se necesita en este presupuesto del concepto
- tiempo que se estima necesario para realizar el concepto

Los insumos particulares deben contar con la siguiente información:

- clave del insumo
- nombre del insumo o descripción corta
- descripción larga o detallada del insumo
- unidades en las que trabajamos el insumo

Los conceptos requieren de la siguiente información:

- clave del concepto
- descripción del concepto (descripción corta)
- unidad en la que se trabaja el concepto
- fecha de actualización de la matriz que forma el concepto
- porcentaje de indirectos que afectan el concepto
- persona que desarrollo el concepto
- iva que afecta al concepto
- precio unitario del concepto
- descripción larga o leyenda desde 50 caracteres hasta 2 hojas
- número de insumos que componen el concepto
- matriz del concepto con : clave del insumo, rendimiento y desperdicio

Para la ruta crítica necesitaremos:

- clave del concepto que forma parte de la red
- duración del concepto
- número de cuadrillas necesarias para cumplir con la duración
- actividades posteriores

En cuanto a los datos generales del proyecto necesitaremos:

- clave del proyecto
- descripción del proyecto
- dependencia a la que se le va a desarrollar el proyecto
- persona responsable en la subdirección
- estado en que se encuentra el proyecto
- fecha de inicio del proyecto, creación de archivos, de presupuesto, de programación, de administración, de conclusión real y de conclusión esperada.
- nombre de los archivos particulares de insumos, conceptos, presupuesto, programación y administración.
- fechas de creación de archivos.

Una vez descritos los archivos, realizaremos las normalizaciones necesarias.

#### Archivo de presupuesto:

- clave del concepto\*
- partida
- volumen
- tiempo

este archivo no necesita ningún tipo de normalización.

#### Archivo de insumos:

- clave del insumo\*
- nombre
- descripción
- unidad

este archivo tampoco necesita ser normalizado.

#### Archivo de conceptos:

- clave del concepto
- descripción corta del concepto
- unidad
- fecha de actualización
- porcentaje de indirectos
- persona que autorizó
- porcentaje de iva
- precio unitario
- leyenda del concepto
- número de insumos

- Los siguientes se repiten n veces
- clave del insumo dentro del concepto
  - rendimiento
  - desperdicio

La primera forma normal nos entregaría el siguiente resultado

*nombre : conceptos1*

elementos:

- clave del concepto\*
- descripción corta
- unidad
- fecha de actualización
- porcentaje de indirectos
- persona que autorizó
- porcentaje de iva
- precio unitario
- leyenda del concepto
- número de insumos

*nombre : conceptos2*

elementos :

- clave del concepto + clave del insumo\*
- rendimiento
- desperdicio

La segunda forma normal quedaría de la misma forma que la primera forma normal.

En la tercera forma normal existe dependencia funcional entre la descripción corta y la descripción larga, pero por manejo se consideró mas conveniente dejarlos dos campos en la misma tabla

Archivo de rule crítica

- clave de la actividad\*
- clave del concepto
- duración de la actividad
- actividad posterior no. 1
- actividad posterior no. 2
- actividad posterior no. 3
- actividad posterior no. 4
- actividad posterior no. 5

este archivo no necesita ser normalizado.

### Archivo de datos generales del proyecto

- clave del proyecto
- descripción
- dependencia
- subdirección
- responsable
- tipo de registro
- clave del estado del proyecto
- descripción del estado del proyecto
- fecha de inicio
- fecha de registro
- fecha de creación de archivos
- fecha de presupuesto
- fecha de programación
- fecha de administración
- fecha de conclusión real
- fecha de conclusión esperada
- archivos de insumos, conceptos, presupuesto, programación y admón
- fecha de creación de estos archivos
- inicio de trabajos relacionados con los archivos.

La primera forma normal quedaría :

*nombre : datos generales 1*

elementos :

- clave del proyecto\*
- descripción
- dependencia
- subdirección
- responsable
- tipo de registro
- clave del estado del proyecto
- descripción del estado del proyecto
- fecha de inicio
- fecha de registro
- fecha de creación de archivos
- fecha de presupuesto
- fecha de programación
- fecha de administración
- fecha de conclusión real
- fecha de conclusión esperada
- clave del archivo de insumos
- clave del archivo de conceptos
- clave del archivo de presupuesto
- clave del archivo de programación
- clave del archivo de administración

*nombre : datos generales 2*

elementos :

- clave obra+clave insumo ó clave concepto ó clave presupuesto ó clave de programación ó clave de administración. \*
- fecha de creación de archivo
- inicio de trabajos

La segunda forma normal quedaria igual a la primera forma normal.

La tercera forma normal nos entregaria:

*nombre : datos generales 1*

elementos :

- clave del proyecto\*
- descripción
- dependencia
- subdirección
- responsable
- tipo de registro
- clave del estado del proyecto
- fecha de inicio
- fecha de registro
- fecha de creación de archivo:
- fecha de presupuesto
- fecha de programación
- fecha de administración
- fecha de conclusión real
- fecha de conclusión esperada
- clave del archivo de insumos
- clave del archivo de conceptos
- clave del archivo de presupuesto
- clave del archivo de programación
- clave del archivo de administración

*nombre : datos generales 2*

elementos :

- clave obra+clave insumo ó clave concepto ó clave presupuesto ó clave de programación ó clave de administración. \*
- fecha de creación de archivo
- inicio de trabajos

*nombre : datos generales 3*

elementos :

- clave del estado del proyecto\*
- descripción del estado del proyecto.



#### 4.1.7 Diccionario de datos del Sistema de Administración Costo de Obra

##### 4.1.7.1 Estructuras de datos del Sistema de Administración Costo de Obra

Los lineamientos bajo los cuales se desarrolló el siguiente capítulo, están detallados en el capítulo titulado "Metodología para el Análisis de Sistemas".

---

#### Archivo: presupuesto

Descripción: Información referente al cálculo de los presupuestos particulares.

---

no.	descripción	tipo	observaciones
1	partida a la que pertenece	string(2)	
2	clave del concepto	string(8)	
3	vol necesario del concepto	real(doble)	
4	tiempo necesario para ejecutar el concepto	entero	

---

organización : indexado por campo 1 y 2

registro\_presupuesto = partida+clave\_concepto+volumen+tiempo

---

---

#### Archivo : insumos particulares

Descripción : características de los registros del archivo de insumos de cada obra

---

no.	descripción	tipo	observaciones
1	clave del insumo	string(8)	
2	nombre del insumo	string(20)	descripción corta
3	descripción detallada del insumo	string(80)	descripción larga
4	unidad en que se trabajen insumo	string(3)	

---

organización : indexado por campo 1

registro\_insumos = clave+nombre+descripción+unidad

---

---

## Archivo: Arch\_conceptos1

Descripción: Aquí tenemos los datos generales que forman parte de cada uno de los conceptos.

---

no	descripción	tipo	observaciones
1	clave del concepto		string(8)
2	descripción corta		string(50)
3	unidad en que se trabaja el concepto		string(3)
4	fecha de actualización de la matriz		string(6)
5	porcentaje de indirectos que afectan al concepto		entero
6	persona que autorizó		string(3)
7	iva que afecta al concepto		entero
8	precio unitario del concepto		real8
9	línea en la que empieza la leyenda		entero
10	carácter en el que empieza la leyenda		entero
11	número de caracteres que forman la leyenda		entero
12	número de insumos que componen al concepto		entero

---

organización: indexado por campo 1 y 2

registro\_conceptos1 = clave\_concepto+descripción\_corta+unidad+fecha+porcentaje\_indirectos+autorizó+iva+p.u.+línea\_leyenda+carácter\_leyenda+no\_caracteres+no\_insumos.

---

## Archivo: Arch\_conceptos2

Descripción: En el siguiente formato se describen las matrices que forman los conceptos

---

no	descripción	tipo	observaciones
1	clave del concepto + clave del insumo		string(16)
2	rendimiento		real
3	desperdicio		real

---

organización: indexado por campo 1

registro\_conceptos2 = cve\_concepto\_cve\_insumo+rendimiento+desperdicio.

---

## Archivo: Arch\_conceptos3

Descripción: Leyenda de los conceptos

---

no.	descripción	tipo	observaciones
1	leyenda del concepto	de 50 caracteres hasta dos hojas	

---

Organización: secuencial

registro\_conceptos3 = 80 caracteres de la leyenda por renglón.

---

## Archivo : ruta crítica

Descripción: la siguiente información esta relacionada con la ruta crítica de cada obra

---

no	descripción	tipo	observaciones
1	clave de la actividad	string(2)	
2	clave del concepto	string(8)	
3	duración de la actividad	real8	
4	no de cuadrillas usadas	entero	
5	actividad posterior no.1	string(2)	
6	actividad posterior no.2	string(2)	
7	actividad posterior no.3	string(2)	
8	actividad posterior no.4	string(2)	
9	actividad posterior no.5	string(2)	

---

organización : secuencial

registro ruta\_crítica : clave\_actividad + clave\_concepto + duración +  
no. de cuadrillas + (actividad posterior)  
\*\*máximo cinco actividades posteriores\*\*

---

## Archivo : datos generales del proyecto

Descripción: información general de cada obra

---

no.	descripción	tipo	observaciones
1	clave del proyecto	string(6)	
2	descripción del proyecto	string(100)	
3	dependencia a quién se le desarrollo	string(5)	
4	subdirección que desarrollo	string(1)	
5	responsable del desarrollo	string(2)	
6	clave de acceso	string(6)	

---

7	tipo de registro	string(2)
8	edo del proy clave	string(2)
9	edo del proy descripción	string(30)
10	fecha de inicio	string(6)
11	fecha de registro	string(6)
12	fecha de creación de archivos	string(6)
13	fecha de presupuesto	string(6)
14	fecha de programación	string(6)
15	fecha de administración	string(6)
16	fecha de termino real	string(6)
17	fecha de termino esperada	string(6)
18	archivo insumos nombre	string(8)
19	archivo insumos fecha	string(6)
20	archivo insumos inicio	string(4)
21	archivo conceptos nombre	string(8)
22	archivo conceptos fecha	string(6)
23	archivo conceptos inicio	string(4)
24	archivo presupuesto nombre	string(8)
25	archivo presupuesto fecha	string(6)
26	archivo presupuesto inicio	string(4)
27	archivo programación nombre	string(8)
28	archivo programación fecha	string(6)
29	archivo programación inicio	string(4)
30	archivo administración nombre	string(8)
31	archivo administración fecha	string(6)
32	archivo administración inicio	string(4)

-----  
organización : indexado por campo 1  
-----

datos\_proyecto: clave+descripción+dependencia+subdirección+responsable  
+clave de seg.+tipo de reg.+edo de proy cve+edo de proy desc  
+fecha de inicio+fecha de registro+fecha de creación de archivos  
+fecha de presupuesto+fecha de programación+fecha de admon  
+fecha de termino real+fecha de termino esperada+nombre archivo  
insumos+archivo de insumos fecha+archivo de insumos inicio

\*\* los últimos tres campos se repiten para conceptos, programación, presupuesto y admon\*\*

#### 4.1.7.2 Flujo de datos del Sistema de Administración Costo de Obra

-----  
Flujo de datos : de ruta crítica a reportes

fuelle : ruta crítica

destino : reportes  
-----

no	descripción	tipo	observaciones
1	manejo de presupuesto particular	archivo	
2	presupuesto particular	archivo	
3	conceptos particulares	archivo	
4	insumos particulares	archivo	

-----

**Nota:**

En general estos son los datos que necesitamos manejar, algunos otros lo describiremos mas a detalle en el documento de diseño.

**4.1.8 Recursos y calendario del Sistema de Administración Costo de Obra**

El presente trabajo fué realizado por un programador- analista con dos años de experiencia, y tardó alrededor de 700 horas/hombre en concluir con el presente escrito, por otro lado fué asesorado por un jefe de departamento de sistemas, el cual invirtió también parte de su tiempo.

En este tipo de proyectos no existen ningún tipo de gastos puesto que el procedimiento es : concertar citas con el usuario, plantear las dudas y seguir con el documento de análisis; no se realizaron viajes y toda la información el usuario la planteó sin problemas.

El calendario de trabajo se presentaría de la siguiente forma:

200 horas documento de análisis

100 horas documento de diseño

400 horas documento de implementación (pruebas y liberación)

**4.1.9 Anexos del Sistema de Administración Costo de Obra**

A continuación se presentan algunos resultados que se entregan actualmente en la Dirección General de Obras.









## **4.2 DISEÑO DEL SISTEMA DE ADMINISTRACION COSTO DE OBRA**

Como lo indica la metodología para el Diseño de sistemas este capítulo tiene los siguientes puntos:

- 4.2.1 Descripción general del Sistema de Administración Costo de Obra**
  - 4.2.1.1 Principales componentes de "software" del Sis. Admon Costo de Obra**
  - 4.2.1.2 Interfaz de "software-hardware" y humanos del Sistema Admon Costo de Obra**
  - 4.2.1.3 Limitaciones y alcances del Sistema de Administración Costo de Obra**
- 4.2.2 Descripción de diseño del Sistema de Administración Costo de Obra**
  - 4.2.2.1 Revisión del D.F.D. del Sistema de Administración Costo de Obra**
  - 4.2.2.2 Revisión de estructuras de datos del Sistema de Admon Costo de Obra**
  - 4.2.2.3 Carta de estructura del "software" del Sistema de Admon Costo de Obra**
- 4.2.3 Descripción de menús del Sistema de Administración Costo de Obra**
  - 4.2.3.1 Carta de estructura de menús del Sistema de Admon Costo de Obra**
  - 4.2.3.2 Descripción de cada menú del Sistema Administración Costo de Obra**

## 4.2 Diseño del sistema de administración costo de obra

### 4.2.1 Descripción general del Sistema Administración Costo de Obra

#### 4.2.1.1 Principales componentes de "software" del Sistema Administración Costo de Obra

El equipo en el que vamos a realizar el sistema es un equipo B-25 de UNISYS el cual presenta la característica de ser una red, en nuestro caso contamos con tres estaciones asociadas y una estación maestra. El sistema operativo que utiliza es conocido con el nombre de BTOS, los comandos están en español y a continuación se enlistan los comandos más importantes como lo pide la metodología para el diseño.

- Bibliotecario. Genera un archivo de biblioteca para poder introducir archivos que se usarán después, por ejemplo todas las formas necesarias deben estar dentro de un archivo de biblioteca.
- Borrar. Elimina archivos.
- Cambiar nombre. Cambia el nombre a los archivos existentes.
- Cobol compile. Compila un archivo fuente en cobol y genera un archivo ejecutable.
- Cobol run. Ejecuta un archivo de cobol previamente compilado.
- Crear archivo. Editor de textos.
- Dmcreate. Genera un programa en dms manager (manejador de la base de datos).
- Dmrun. Ejecuta un programa creado con dmcreate.
- Editar archivo. Es una segunda forma de editar.
- Ejecutar. Ejecutar un programa hecho en pascal previamente compilado y ligado.
- Enlazar. Liga uno o más programas en pascal y entrega de salida un archivo ejecutable.
- Enhaced multiplan. Hoja de cálculo basada en multiplan.
- Formis editor. Editor de formas de equipos B-25.
- Freport. Despliega la información referente a alguna forma, como por ejemplo número de campos, tamaño de la forma en bits, tamaño de los campos en caracteres, etc.
- Imprimir. Lanza a la impresora el archivo que nosotros indiquemos.
- ISAM borrar. Elimina archivos ISAM, tanto el archivo de índices como el archivo de la información.
- ISAM copiar. Copia la información y los índices de un archivo ISAM hacia otro.
- ISAM crear. Crea el cascarón de un archivo ISAM.
- ISAM estado. Emite información del archivo, como número de registros, tamaño del registro, tamaño de la llave, etc.
- ISAM reorganizar. Borra el anterior archivo de índices de un archivo ISAM y genera un nuevo archivo de índices.
- Pascal. Compila un archivo fuente escrito en pascal.
- Ruta. Utilizado para cambiarnos de directorio.
- Swp. Procesador de palabras.

En cuanto a los lenguajes que maneja el equipo tenemos los siguientes:

- Cobol. Es un cobol normal con manejo de archivos ISAM, en este caso tiene la ventaja de que puede jalar formas diseñadas por nosotros externamente, este cobol es interprete.
- Pascal. Es un lenguaje estructurado, el pascal que maneja esta máquina tiene la ventaja de utilizar archivos indexados, jalar formas de las bibliotecas y también un uso

especial de archivos secuenciales.

-Fortran. Tiene las mismas características que el pascal, solo que al parecer no se dispone de esta utilidad.

Refiriendonos a paquetes complementarios, solamente tendremos una hoja de cálculo conocida como multiplan y un procesador de palabras que no proporciona muchas facilidades llamado "Secretary word process" (SWP).

Los equipos B-25 utilizan un manejador de base de datos llamado "Data Manager" el cual no es muy poderoso para el desarrollo de sistemas, pero nos da una facilidad muy grande con respecto al manejo de la información en lo que se refiere a altas, bajas, cambios y consultas de archivos ISAM.

#### 4.2.1.2 Interfases de "hardware-software" y humanos del Sistema Administración Costo de Obra

En cuanto a la interfase "software"-humanos, podemos decir que todo nuestro sistema será a base de menús para facilitar el manejo del sistema, además de que se preguntará en todos los casos si la tecla de ayuda fué accionada, en caso de haber sido accionada aparecerán mensajes con el tipo de acciones que se pueden tomar. Por último el "software" validará toda la información que sea accesada al sistema, en el caso de las fechas utilizaremos el formato dd/mm/aa, en el caso de los números enteros se validará que sean números del 0 al 9 por dígito y en el caso de los campos reales se partirán en dos números, uno antes del punto y otro después del punto para validarlos como números enteros.

Refiriendonos a la interfase "hardware"-humanos, en este caso lo mejor es usar una terminal del equipo B-25 para la captura y auxiliarnos de las teclas con las que cuenta el teclado de esta terminal, por ejemplo utilizaremos la tecla de fin, cncl, return, flecha arriba, flecha abajo, flecha izquierda, flecha derecha, etc.

#### 4.2.1.3 Limitaciones y alcances del diseño del Sistema Administración Costo de Obra

En cuanto a los alcances del diseño los podemos reducir en los siguientes puntos:

- realizar la gráfica de la ruta crítica
- volver mas eficiente los procesos administrativos
- disminuir la carga de trabajo del personal que interviene en el proceso administrativo
- disponer de la información de las obras en forma instantánea
- tener cantidades de material a utilizar en obra, o en su defecto conocer la cantidad de dinero a gastar
- cambiar tiempos y volver a graficar la ruta crítica cuantas veces sea necesario
- ayudar en la toma de decisiones
- entregar todos los cálculos necesarios en una ruta crítica.

Estos serían los puntos más importantes en cuanto a los alcances del diseño; refiriendonos a las limitaciones podemos comentar que la ruta crítica se ve mejor en una gráfica y que si esta gráfica fuera hecha en un plotter con diferentes colores resaltando ruta crítica, holguras, iniciación, etc., la claridad de los resultados sería muy grande, por

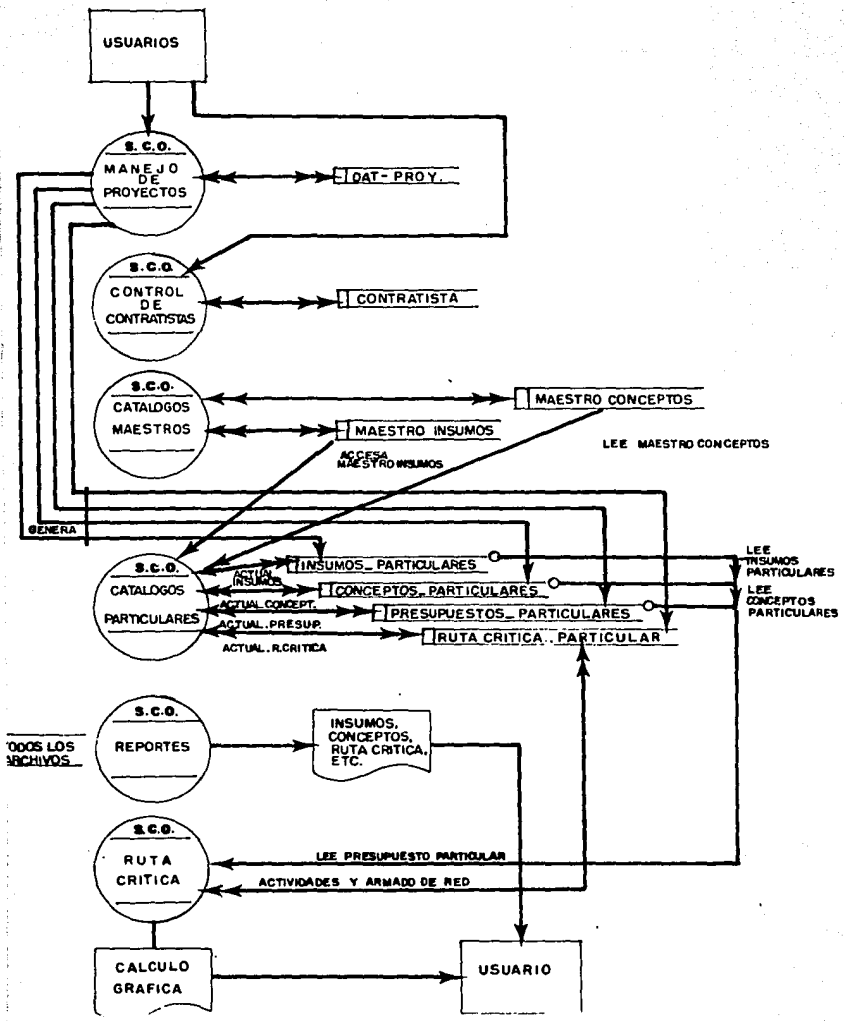
SALA DE LA DIRECTORIA  
1986  
1986

esto consideramos que no contar con un plotter es una limitación.

#### 4.2.2 Descripción del diseño del Sistema Administración Costo de Obra

##### 4.2.2.1 Revisión del diagrama de flujo de datos del Sistema Administración Costo de Obra.

Después de haber revisado el diagrama de flujo de datos presentado en el capítulo titulado "Análisis del sistema de administración costo de obra", se plantea a continuación el diagrama de flujo de datos con el que trabajaremos en el desarrollo del sistema.



#### 4.2.2.2 Revisión de estructuras de datos del Sistema de Administración Costo de Obra

Aquí como lo indica el capítulo de metodología del diseño, debemos volver a realizar las normalizaciones de nuestros archivos; a continuación realizaremos estas normalizaciones y dejaremos lista la información a utilizar en nuestro sistema.

Información para el manejo del presupuesto.

- clave del concepto
- partida
- volumen
- tiempo

Después de verificar las tres formas normales, el archivo quedaría:

**Nombre:** Arch\_presupuesto

- clave del concepto\*
- partida
- volumen
- tiempo

El formato de estructura de datos quedaría:

-----  
Archivo: presupuesto

Descripción: información referente al cálculo de los presupuestos particulares.

no.	descripción	tipo	observaciones
1	partida a la que pertenece	string(2)	
2	clave del concepto	string(8)	
3	vol necesario del concepto	real(8)	
4	tiempo necesario para ejecutar el concepto	entero	

-----  
organización: indexado por campo 1 y 2  
-----

Información para el manejo de los conceptos

- clave del concepto
- descripción del concepto (descripción corta)
- unidad en la que se trabaja el concepto
- fecha de actualización de la matriz que forma el concepto
- porcentaje de indirectos que afecta el concepto
- persona que desarrollo el concepto
- iva que afecta el concepto
- precio unitario del concepto
- descripción larga o leyenda desde 50 caracteres hasta dos hojas
- número de insumos que componen el concepto
- matriz del concepto con: clave del insumo, rendimiento y desperdicio.

La primera forma normal nos entregaría el siguiente resultado

**Nombre:** Arch\_conceptos1

- clave del concepto\*
- descripción corta
- unidad
- fecha de actualización
- porcentaje de indirectos
- persona que autorizó
- porcentaje de iva
- precio unitario
- leyenda del concepto
- número de insumos

**Nombre:** Arch\_conceptos2

- clave del concepto + clave del insumo\*
- rendimiento
- desperdicio

La segunda forma normal quedaría igual a la primera forma normal.

La tercera forma normal entregaría el siguiente resultado, como existe dependencia funcional entre la descripción corta y la descripción larga o leyenda, la leyenda la colocaremos en otro archivo, pero por las características de estas leyendas, usaremos un archivo secuencial y con apuntadores conoceremos las posiciones de inicio y fin de estas leyendas.

Los archivos quedarían después de la tercera forma normal de la siguiente forma:

**Nombre:** Arch\_conceptos1

- clave del concepto\*
- descripción corta
- unidad
- fecha de actualización
- porcentaje de indirectos
- persona que autorizó
- porcentaje de iva
- precio unitario
- línea en la que empieza la leyenda
- caracter en el que empieza la leyenda
- número de caracteres que forman la leyenda
- número de insumos

**Nombre:** Arch\_conceptos2

- clave del concepto+clave del insumo\*
- rendimiento
- desperdicio

**Nombre:** Arch\_conceptos3

- leyendas consecutivas

Los formatos de estructura de datos quedarían:

Archivo: Arch\_conceptos1

Descripción: Aquí tenemos los datos generales que forman parte de cada uno de los conceptos.

no	descripción	tipo	observaciones
1	clave del concepto		string(8)
2	descripción corta		string(50)
3	unidad en que se trabaja el concepto		string(3)
4	fecha de actualización de la matriz		string(6)
5	porcentaje de indirectos que afectan al concepto		entero
6	persona que autorizó		string(3)
7	iva que afecta al concepto		entero
8	precio unitario del concepto		real5
9	línea en la que empieza la leyenda		entero
10	carácter en el que empieza la leyenda		entero
11	número de caracteres que forman la leyenda		entero
12	número de insumos que componen el concepto		entero

organización: indexado por campo 1 y 2

Archivo: Arch\_conceptos2

Descripción: En el siguiente formato se describen las matrices que forman los conceptos:

no	descripción	tipo	observaciones
1	clave del concepto + clave del insumo		string(16)
2	rendimiento		real
3	desperdicio		real

organización: indexado por campo 1

Archivo: Arch\_conceptos3

Descripción: Leyenda de los conceptos

no.	descripción	tipo	observaciones
1	leyenda del concepto		de 50 caracteres hasta dos hojas

Organización: secuencial



### **Características especificadas para los conceptos**

Por convención se acepta dividir este módulo en dos tipos de conceptos: al primero se le denomina conceptos básicos y al segundo conceptos determinados.

El primer tipo, conceptos básicos, comprende a los que por su frecuencia de uso, prácticamente tienen un carácter de insumo y para facilitar su incorporación en los análisis de costo de los conceptos determinados de obra se les concede el carácter de particularidad.

En la clave de catalogación numérica con capacidad para ocho dígitos se conviene para su lectura e interpretación de izquierda a derecha que los dos primeros siendo 00 indicarán que estamos hablando de un concepto básico.

Los dos siguientes dígitos relacionan al concepto básico con el tipo de insumo preponderante, siguiendo la convención adoptada para el módulo de insumos.

Los dígitos cinco y seis indicarán características genéricas.

Las particularidades de los conceptos básicos en sus enunciados precisarán claramente sus características propias y en la clave de catalogación se identificarán con los dígitos colocados a la extrema derecha.

El segundo tipo de conceptos, serán los conceptos determinados de obra y que se refieren a las particularidades propiamente dichas de los conceptos de obra.

La clave numérica de catalogación que se adopta, tiene una amplitud de ocho dígitos, de los cuales en su lectura de izquierda a derecha los seis primeros dígitos invariablemente corresponderán con los indicadores en las especificaciones generales de construcción en la UNAM, y los dos últimos de la extrema derecha se conviene reservarlos para precisar la particularidad característica del concepto de obra.

En el caso de los primeros seis dígitos de la clave se deberán desarrollar de acuerdo a lo siguiente:

- División de la inversión de obra en partidas generales, en la clave se identificará por el primer dígito.
- División de partidas generales en subpartidas de obra, en la clave se identifican por el segundo dígito de izquierda a derecha.
- División de subpartidas en grupos de obra, en la clave de catalogación se identifican por el tercer dígito de izquierda a derecha.
- División de grupos en subgrupos de obra, en la clave de catalogación estarán representados por los dígitos cuatro, cinco y seis de izquierda a derecha.

#### **Información necesaria para la ruta crítica**

- clave del concepto que forma parte de la red
- duración del concepto
- número de cuadrillas necesarias para cumplir con la duración
- actividades posteriores

La primera forma normal quedaría de la siguiente forma:

**Nombre:** Arch\_rutacritica

- clave de la actividad\*
- clave del concepto
- duración de la actividad
- actividad posterior #1
- actividad posterior #2
- actividad posterior #3
- actividad posterior #4
- actividad posterior #5

Después de revisar hasta la tercera forma normal, definimos que este archivo queda igual en la segunda y en la tercera forma normal.

El formato de estructura de datos quedaría de la siguiente manera:

**Archivo:** Arch\_rutacritica

**Descripción:** La siguiente información está relacionada con la ruta crítica de cada obra.

no	descripción	tipo	observaciones
1	clave de la actividad	string(2)	
2	clave del concepto	string(8)	
3	duración de la actividad	real8	
4	no de cuadrillas usadas	entero	
5	actividad posterior no. 1	string(2)	
6	actividad posterior no. 2	string(2)	
7	actividad posterior no. 3	string(2)	
8	actividad posterior no. 4	string(2)	
9	actividad posterior no. 5	string(2)	

**organización:** secuencial

**Información general para el manejo de insumos particulares**

- clave del insumo
- nombre del insumo o descripción corta
- descripción larga o detallada del insumo
- unidades en las que trabajamos el insumo

La primera forma normal quedaría de la siguiente manera:

**Nombre:** Arch\_insumos

- clave insumo\*
- descripción corta
- descripción larga
- unidad

La segunda forma normal quedaría igual a la primera forma normal.

La tercera forma normal nos indicaría que existe dependencia funcional entre el campo de descripción corta y el campo de descripción larga, no los separamos porque la descripción corta no es una clave y no se puede tomar como llave para otro archivo, por esto el archivo quedaría igual.

El formato de estructura de datos quedaría de la siguiente manera

Archivo: Arch\_insumos

Descripción: Características generales de los insumos particulares, insumos que forman parte de una obra.

no.	descripción	tipo	observaciones
1	clave del insumo	string(8)	
2	nombre del insumo	string(20)	descripción corta
3	descripción detallada del insumo	string(80)	descripción larga
4	unidad en que se trabaja insumo	string(3)	

organización : indexado por campo 1

Los insumos particulares se trabajarán con las siguientes convenciones:

1. La clave de catalogación de tipo numerico indicará en su lectura de izquierda a derecha, con dos dígitos el tipo de insumo, los dos siguientes dígitos el grupo del insumo, los dos siguientes el subgrupo de insumo y los dos últimos dígitos las características particulares del insumo.
2. En el nombre, el insumo será descrito en forma sintética y precisa.
3. En la descripción, el insumo será descrito a detalle.
4. La unidad de medición será representada en forma abreviada de acuerdo a la forma usual.

Respecto a la información que debe ir en contemplada en la clave se seguirá la siguiente convención:

- 01 materiales
- 02 obra de mano
- 03 maquinaria
- 04 partidas porcentuales
- 05 contratos complementarios
- 06 deshecho

Información necesaria para el manejo de los proyectos

- clave del proyecto
- descripción
- dependencia para la que se desarrollo
- subdirección responsable
- persona responsable
- tipo de registro
- clave del estado del proyecto

- descripción del estado del proyecto
- fecha de inicio
- fecha de registro
- fecha de creación de archivos
- fecha de presupuesto
- fecha de programación
- fecha de administración
- fecha de conclusión real
- fecha de conclusión esperada
- nombre del archivo de insumos, conceptos, presupuesto, programación y administración
- fecha de creación de estos archivos
- inicio de trabajos relacionados con los archivos

La primera forma normal nos quedaría de la siguiente manera

**Nombre: Datos\_generales1**

- clave del proyecto\*
- descripción
- dependencia
- subdirección
- responsable
- tipo de registro
- clave del estado del proyecto
- descripción del estado del proyecto
- fecha de inicio
- fecha de registro
- fecha de creación de archivos
- fecha de presupuesto
- fecha de programación
- fecha de administración
- fecha de conclusión real
- fecha de conclusión esperada
- clave del archivo de insumos
- clave del archivo de conceptos
- clave del archivo de presupuesto
- clave del archivo de programación
- clave del archivo de administración

**Nombre: Datos\_generales2**

- clave\_obra + clave\_insumos ó clave de conceptos ó clave de presupuesto ó clave de programación ó clave de administración\*
- fecha de creación de archivos
- inicio de trabajos respecto a estos archivos

La segunda forma normal quedaría igual a la primera forma normal.

La tercera forma normal nos entregaría el siguiente resultado:

**Nombre: Datos\_generales1**

- clave del proyecto\*
- descripción
- dependencia
- subdirección
- responsable

- tipo de registro
- clave del estado del proyecto
- fecha de inicio
- fecha de registro
- fecha de creación de archivos
- fecha de presupuesto
- fecha de programación
- fecha de administración
- fecha de conclusión real
- fecha de conclusión esperada
- clave del archivo de insumos
- clave del archivo de conceptos
- clave del archivo de presupuesto
- clave del archivo de programación
- clave del archivo de administración

**Nombre: Datos\_generales2**

- clave obra + clave insumos ó clave de conceptos ó clave de presupuesto ó clave de programación ó clave de administración\*
- fecha de creación de archivos
- inicio de trabajos

**Nombre: Datos\_generales3**

- clave del estado del proyecto\*
- descripción del estado del proyecto

El formato de estructura de datos quedaría de la siguiente manera:

-----  
 Archivo: Datos\_generales1

Descripción: Información general de cada obra

no.	descripción	tipo	observaciones
1	clave del proyecto		string(6)
2	descripción del proyecto		string(100)
3	dependencia a quién se le desarrollo		string(5)
4	subdirección que desarrollo		string(1)
5	persona responsable del desarro		string(2)
6	tipo de registro		string(2)
7	edo del proy clave		string(2)
8	fecha de inicio		string(6)
9	fecha de registro		string(6)
10	fecha de creación de archivos		string(6)
11	fecha de presupuesto		string(6)
12	fecha de programación		string(6)
13	fecha de administración		string(6)
14	fecha de termino real		string(6)
15	fecha de termino esperada		string(6)
16	archivo insumos clave		string(8)
17	archivo conceptos clave		string(8)
18	archivo presupuesto clave		string(8)

19	archivo programación clave	string(8)
20	archivo administración clave	string(8)

-----  
 organización : indexado por campo 1  
 -----

-----  
**Archivo: Datos\_generales2**

Descripción: Tabla de los archivos de insumos, conceptos, presupuesto, administración y programación

no.	descripción	tipo	observaciones
1	clave obra + clave insumo ó clave concepto ó clave presupuesto ó clave de admón ó clave de programación		
2	fecha de creación de archivos	string(14)	
3	inicio de trabajos	string(6)	string(6)

-----  
 organización: indexado por el campo 1  
 -----

-----  
**Archivo: Datos\_generales3**

Descripción: Tabla de los estados del proyecto

no.	descripción	tipo	observaciones
1	clave del estado del proyecto	string(2)	
2	descripción del edo. del proy.	string(50)	

-----  
 organización: indexado por el campo 1  
 -----

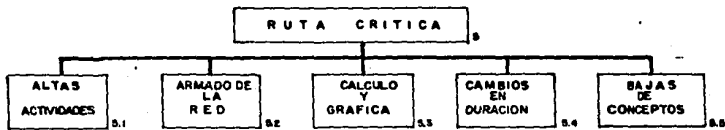
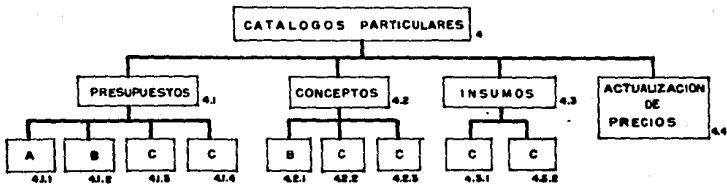
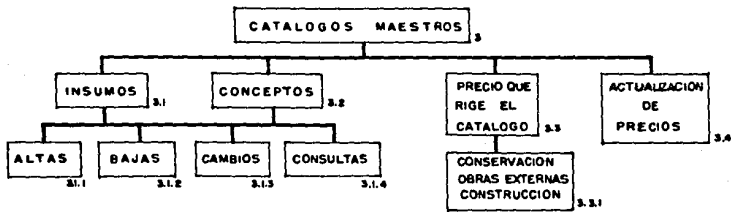
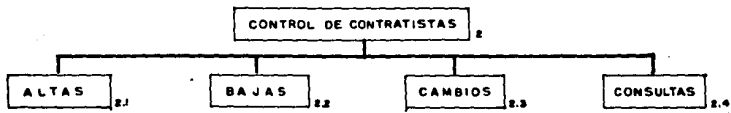
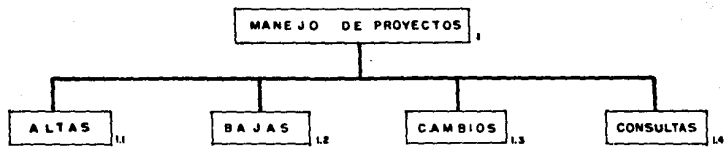
#### 4.2.2.3 Carta de estructura de software del sistema de administración costo de obra.

Para desarrollar este punto nos basamos en la teoría expuesta en el capítulo llamado metodología para el diseño de sistemas, en la parte de carta de estructura de software; esta teoría la aplicamos al diagrama de flujo de datos y nos entrega el siguiente resultado.

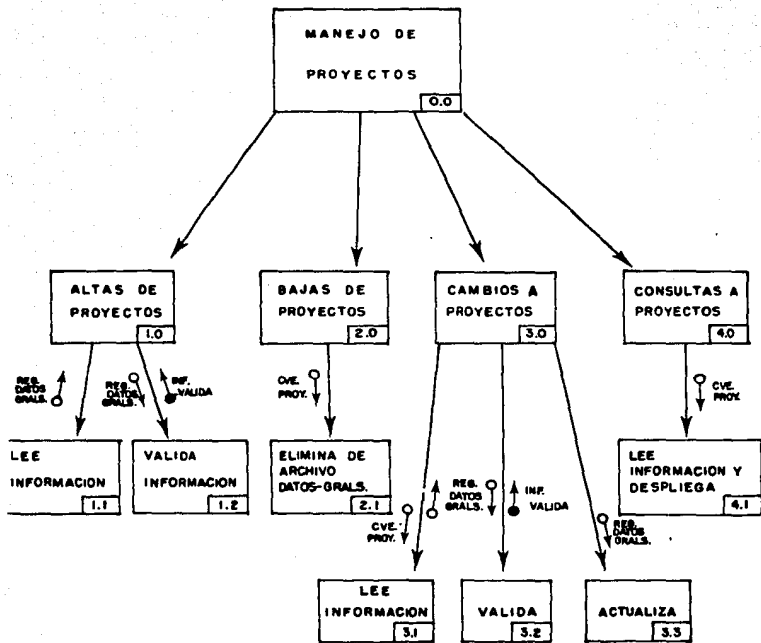
Dentro de nuestro diagrama de flujo de datos corregido, tenemos seis módulos los cuales desarrollamos aquí.

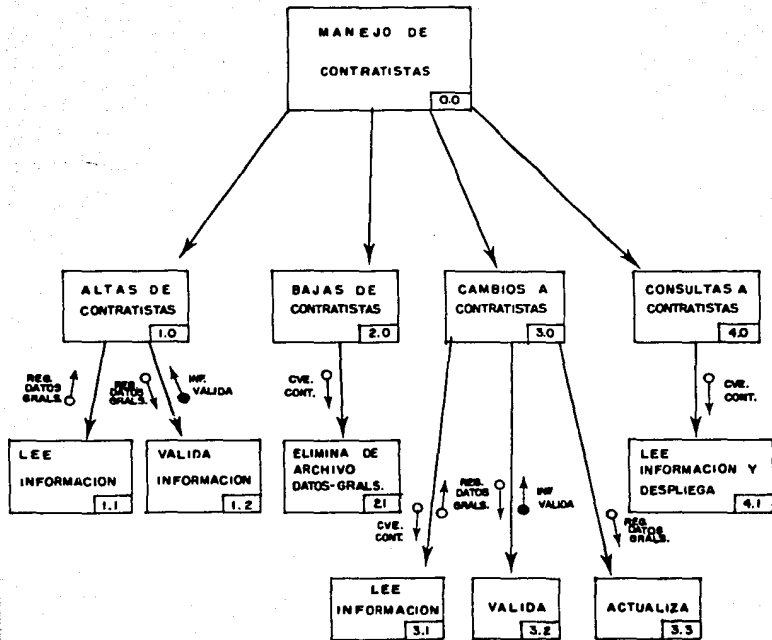
## **MENU PRINCIPAL**

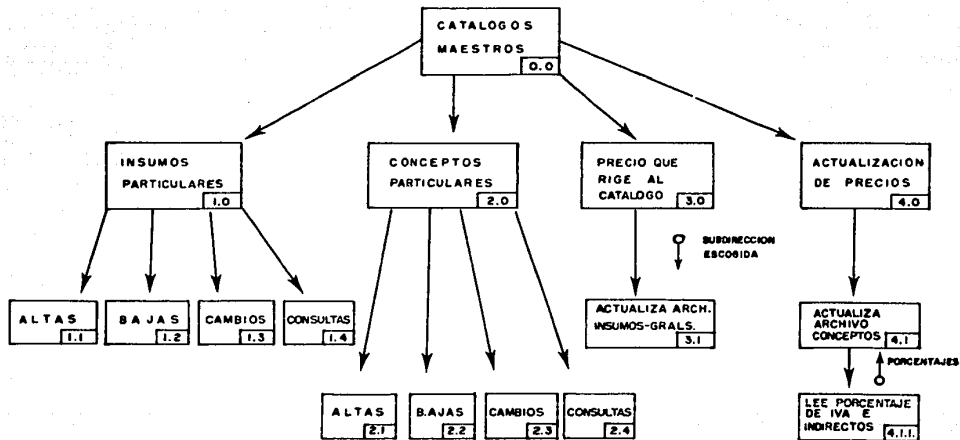
- 1. MANEJO DE PROYECTOS**
- 2. CONTROL DE CONTRATISTAS**
- 3. CATALOGOS MAESTROS**
- 4. CATALOGOS PARTICULARES**
- 5. REPORTES**
- 6. RUTA CRITICA**





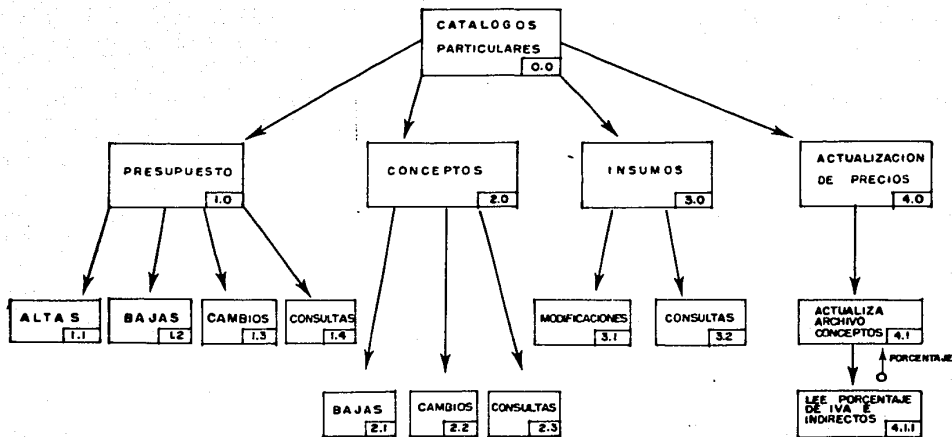






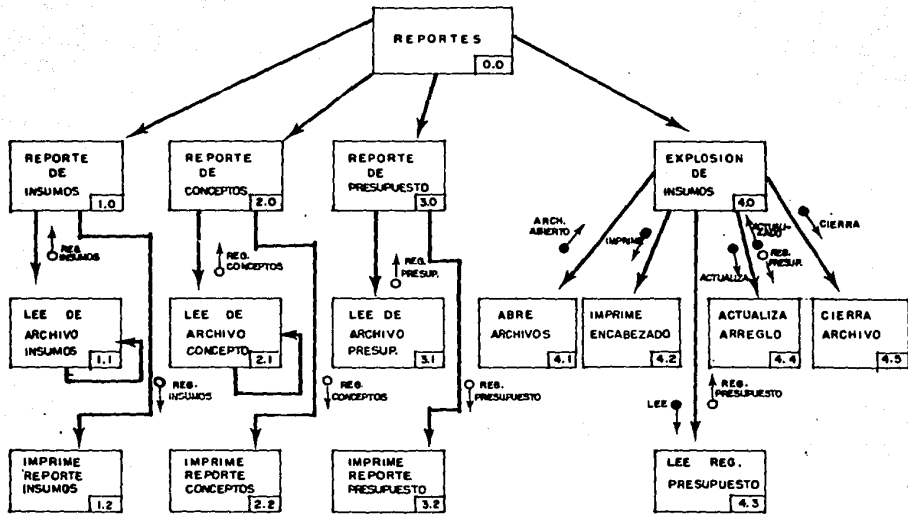
**NOTA:**

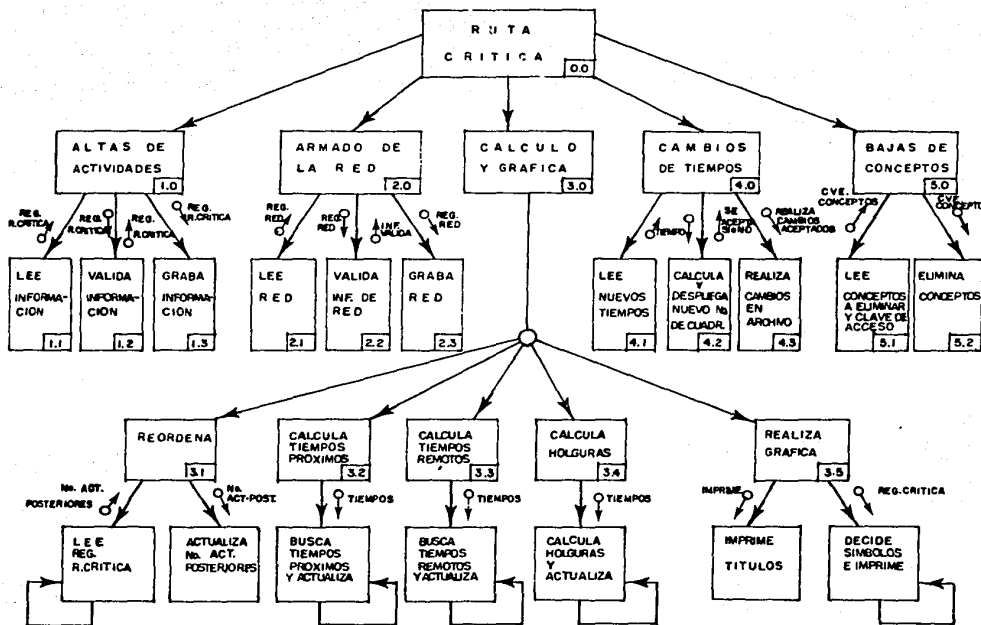
TODOS LOS MODULOS DE ALTAS,BAJAS,CAMBIOS Y CONSULTAS DEBEN DESARROLLARSE IGUAL QUE SE DESARROLLARON ESTOS MISMOS MODULOS EN EL DIAGRAMA DE MANEJO DE PROYECTOS.



**NOTA:**

TODOS LOS MODULOS DE ALTAS, BAJAS, CAMBIOS Y CONSULTAS DEBEN DESARROLLARSE IGUAL QUE SE DESARROLLAN ESTOS MISMOS MODULOS EN EL DIAGRAMA DE MANEJO DE PROYECTOS.





#### **4.2.3 Descripción de menús del Sistema Administración Costo de Obra**

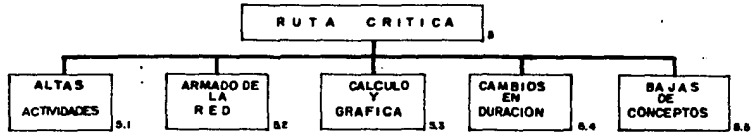
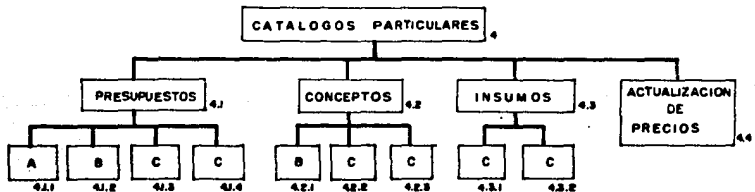
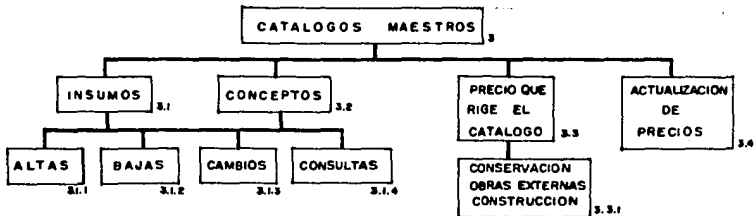
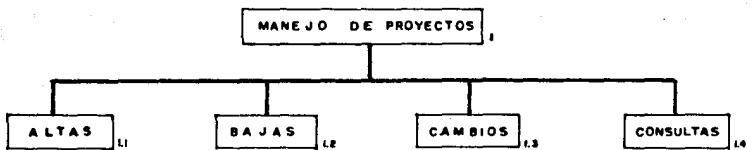
##### **4.2.3.1 Carta de estructura de menús del Sistema de Administración Costo de Obra**

Como se planteó en el documento de diseño, propondremos aquí los diferentes menús que tendrá nuestro sistema, en este caso quedaría de la siguiente manera.

# **M E N U   P R I N C I P A L**

- 1.     M A N E J O   D E   P R O Y E C T O S**
- 2.     C O N T R O L   D E   C O N T R A T I S T A S**
- 3.     C A T A L O G O S   M A E S T R O S**
- 4.     C A T A L O G O S   P A R T I C U L A R E S**
- 5.     R E P O R T E S**
- 6.     R U T A   C R I T I C A**





#### 4.2.3.2 Descripción de cada menú del Sistema de Administración Costo de Obra

Nuestro primer menú o menú principal tendrá la posibilidad de elegir alguna de las siguientes opciones:

1. Manejo de proyectos
2. Control de proveedores
3. Catálogos maestros
4. Catálogos particulares
5. Reportes
6. Ruta crítica

1. Manejo de proyectos. Maneja las características generales de una obra como son: fecha de inicio, descripción, fecha de creación de archivos, dependencia para la que se realiza, etc., se tienen las siguientes opciones:

- 1.1. Altas. Cuando se necesita comenzar una nueva obra, se recurrirá a este módulo, aparecerá un formato el cual nos pedirá la información general de la obra.
- 1.2. Bajas. Con dar la clave de la obra esta podrá ser eliminada, como es muy delicado eliminar obras completas, contaremos con un implemento de seguridad para proteger las obras.
- 1.3. Modificaciones. En caso de que ocurra algún error en la captura o se necesite algún cambio de la información, podremos utilizar este módulo.
- 1.4. Consultas. Este módulo es sumamente útil, puesto que nos permitirá verificar la información, tener la información a la mano para la toma de decisiones, también nos permitirá tener bastante información en cuestión de segundos en caso de solo necesitar una simple consulta.

2.- Control de proveedores. Manejará toda la información referente a los proveedores y contratistas con los que trabaja la D.G.O.S.G., esta información comprende, clave del proveedor, productos que suministra, dirección, persona a contactar, etc., las opciones son:

- 2.1. Alta. Si es necesario introducir un nuevo proveedor en nuestro archivo recurrimos a este módulo, accedendo toda la información del proveedor.
- 2.2. Bajas. Tecleando la clave del proveedor que queremos eliminar de nuestro archivo este se eliminará.
- 2.3. Modificaciones. En caso de querer cambiar algún dato de un proveedor utilizaremos este punto, lo único que no se puede cambiar es la clave del proveedor y el nombre.
- 2.4. Consultas. Para conocer las características del proveedor, nombre, dirección, etc. utilizaremos este módulo.

Existirán dos tipos de catálogos, los primeros se denominarán catálogos maestros y existirá uno para insumos y uno para conceptos. Los segundos serán los catálogos particulares y serán referidos a cada obra, nuestros catálogos particulares serán de insumos y conceptos, aquí se incluye al manejo del archivo de presupuesto por ser también un presupuesto por obra.

3.- Catálogos maestros. Maneja toda la información de los catálogos maestros de insumos y de conceptos, así como el precio que registró en el catálogo de conceptos y los actualizaciones de precios, se manejen las siguientes opciones:

- 3.1. Insumos. El catálogo maestro de insumos es manejado en este punto, para su manejo se deberán desarrollar los correspondientes módulos de altas, bajas, cambios y consultas.
- 3.2. Conceptos. Los conceptos maestros se podrán manejar en este punto, para hacer este manejo se deben de tener los siguientes módulos: altas, bajas, cambios y consultas.
- 3.3. Precio que rige el catálogo. El sistema maneja cuatro diferentes precios de insumos, un precio por cada subdirección (Conservación, Construcción y Obras externas) y un último precio denominado precio D.G.O., siendo este el precio que rige para cualquier cálculo posterior, en caso que se necesite que el precio de alguna subdirección domine en los futuros cálculos se recurre a este módulo, las opciones deberán ser por tanto.
  - 3.3.1. Conservación
  - 3.3.2. Obras externas
  - 3.3.3. Construcción

Siendo que al elegir alguna opción, los precios de esta subdirección serán copiados al espacio designado para precios D.G.O.

- 3.4. Actualizaciones de precios. Dentro de nuestro catálogo maestro de conceptos, este precio puede ser afectado por un porcentaje de indirectos y por un porcentaje de I.V.A. que pueden variar, estos dos porcentajes se teclearán aquí y automáticamente se recalcularán los precios de los conceptos afectados por los nuevos porcentajes.

4. Catálogos particulares. Estos catálogos particulares son los que pertenecen a cada una de las obras que se dieron de alta en el punto uno "Catálogo costo de obra" y se tendrán para su manejo las siguientes opciones:

- 4.1. Presupuesto. Dentro de una obra en particular, uno de los puntos que nos interesa tener controlado es el presupuesto de la misma, para llevar a cabo este control necesitaremos los siguientes módulos.
  - 4.1.1. Altas. Cuando nuestro presupuesto es nuevo o si queremos meter nuevos conceptos recurriremos a este punto.
  - 4.1.2. Bajas. En caso de querer eliminar conceptos que ya forman parte de nuestro presupuesto utilizaremos este módulo.
  - 4.1.3. Cambios. Si alguno de los conceptos que forman nuestro presupuesto debe ser cambiado en cantidad, nombre, etc., se recurrirá a este módulo.
  - 4.1.4. Consultas. Módulo usado para consultar los conceptos que forman nuestro presupuesto, el subtotal por partida, el total del presupuesto, etc.
- 4.2. Conceptos. Aquí podremos manejar todo lo relacionado con los conceptos particulares de una obra, no se podrá dar de alta un concepto porque las altas se realizan sólo cuando damos de alta los conceptos en el presupuesto, pero si podemos realizar bajas, cambios y consultas, para hacer esto se deberán desarrollar los módulos correspondientes.
- 4.3. Insumos. Los insumos particulares podrán ser modificados y consultados en este punto, por seguridad no se podrán dar de bajas y se podrán dar de alta al mismo tiempo que se realiza el alta de los conceptos en el presupuesto

- 4.4. Actualizaciones de precios. Este punto llevará a cabo las mismas acciones que el punto 3.4. solamente que aquí efectuaremos un catálogo de conceptos particular y en el 3.4. se efectará el catálogo de conceptos maestro.
- 5.- Reportes. Todos los reportes que se soliciten, siempre y cuando sean viables, se incluirán aquí. Podremos mencionar los más sobresalientes. Reporte de insumos, reporte de conceptos, reporte de presupuesto, explosión de insumos, etc.
- 6.- Ruta crítica. En caso de necesitar el cálculo y la gráfica de la ruta crítica de una obra en particular, entraremos a este módulo, el cual nos presentará las siguientes opciones:
  - 6.1. Atlas de actividades. Aquí indicaremos cuáles de las actividades que forman parte del presupuesto particular, integrarán nuestra ruta crítica. Para facilidad en el manejo de la ruta ligaremos la clave de la actividad de 9 dígitos con una clave del tipo letra-letra de dos dígitos.
  - 6.2. Armado de la red. La ruta crítica se obtiene de una red de actividades, para poder armar esta red usaremos este punto, señalando la actividad que necesitaremos y las actividades anteriores y posteriores a esta.
  - 6.3. Cálculo y gráfica. Una vez armada la red de actividades y obtenidos los tiempos de duración de las actividades del catálogo de conceptos, se puede realizar el cálculo de la ruta crítica, con el cálculo hecho sólo nos faltaría graficar, estos puntos se obtendrán aquí.
  - 6.4. Cambio de tiempos. Si los tiempos de duración de las actividades son muy grandes estos se pueden disminuir aumentando cuadrillas, esta operación se realiza en este punto.
  - 6.5. Bajas de conceptos. En caso de que algún concepto esté mal dado o ya no se realice, se puede recurrir a este módulo. Como es muy delicado en este caso el manejo de las bajas, se requerirá de una clave.

#### **4.3 IMPLEMENTACION DEL SISTEMA DE ADMINISTRACION COSTO DE OBRA**

Como lo indica la metodología para la implementación de sistemas este capítulo contiene los siguientes puntos:

- 4.3.1 Características generales de la implementación del Sistema de Admon Costo de Obra**
  - 4.3.1.1 Elección del "software" para el Sistema de Administración Costo de Obra**
  - 4.3.1.2 Tiempos de programación del Sistema de Administración Costo de Obra**
  - 4.3.1.3 Comentarios sobre la liberación del Sistema de Administración Costo de Obra**
- 4.3.2 Probando el "software" del Sistema de Administración Costo de Obra**
  - 4.3.2.1 Pesos en las pruebas del "software" del Sistema de Admon Costo de Obra**
    - 4.3.2.1.1 Pruebas de unidad del Sistema de Administración Costo de Obra**
    - 4.3.2.1.2 Pruebas de integración del Sistema de Administración Costo de Obra**
    - 4.3.2.1.3 Pruebas de validación del Sistema de Administración Costo de Obra**
    - 4.3.2.1.4 Probando las funciones del Sistema de Administración Costo de Obra**
- 4.3.3 Mantenimiento del "software" del Sistema de Administración Costo de Obra**
  - 4.3.3.1 Mantenimiento más fácil desde el desarrollo del Sis. de Admon Costo de Obra**
  - 4.3.3.2 Aspectos administrativos del mantenimiento del Sis. de Admon Costo de Obra**
  - 4.3.3.3 Problemas durante el mantenimiento del Sistema de Admon Costo de Obra**

#### 4.3.1 Características generales de la implementación del Sistema de Administración Costo de Obra

##### 4.3.1.1 Elección del "software" para el Sistema de Administración Costo de Obra

Dentro del capítulo titulado "Diseño del Sistema Administración Costo de Obra" en el punto 4.2.1.1 " Principales Componentes de Software del Sistema de Administración Costo de Obra" explicamos cuáles son los elementos de "software" que nos ofrecen los equipos B-25, basándonos en estas características planteamos a continuación el "software" que nosotros consideramos el adecuado para el desarrollo del sistema:

Primeramente como lenguaje de programación utilizaremos el pascal que nos ofrecen los equipos B-25, la elección del pascal se basó en que es un lenguaje estructurado a diferencia del cobol, tiene un manejo de archivos tan bueno como el manejo de archivos en cobol, se pueden hacer formas diseñadas externamente en una utilería llamada editor de formas; por todas estas características nos inclinamos a utilizar el pascal como lenguaje de programación.

En segundo término, se utilizará el editor de formas con el que se cuenta puesto que es una utilería muy práctica y de fácil manejo, además de que dará mayor vistosidad al sistema.

Estos dos son las principales herramientas de "software" que utilizaremos en el desarrollo del sistema, cabe señalar que tal vez se usen algunas otras herramientas de "software" pero como su uso no se vislumbra en este momento no se explican.

Para poder utilizar el pascal debemos usar cuatro comandos de sistema operativo.

- Editor. El cual nos permite capturar nuestro código fuente.
- Pascal. Convierte un archivo en pascal y convierte el archivo fuente en archivo objeto.
- Ligar. Convierte uno o varios archivos objeto en un archivo ejecutable.
- Ejecutar. Ejecuta un programa hecho en pascal.

Para generar una forma utilizamos los siguientes comandos:

- Forms editor. Nos permite crear una forma, esta utilería no cuenta con ayuda en línea, pero el manual es muy claro en su forma de uso.
- Freport. Pide el nombre de la forma generada en forms editor y nos indica el tamaño de la forma para poderla hacer desde el programa fuente en pascal.

##### 4.3.1.2 Tiempos de programación del Sistema de Administración Costo de Obra

Como lo indica el capítulo de " metodología para la implementación " nos basaremos en los pseudocódigos propuestos para definir estos tiempos de programación.

módulo de explosión de insumos	80 horas
etapas de actividad a la ruta crítica	40 horas
armado de la red	40 horas

bajas de actividades	24 horas
cambios de duración	80 horas
procesa ruta crítica	80 horas
realiza gráfica de la ruta crítica	80 horas

Cabe señalar que los tiempos aquí presentados se entienden como horas trabajadas, no se toman en cuenta tiempos muertos.

#### 4.3.1.3 Comentarios sobre la liberación del Sistema de Administración Costo de Obra

El "software" desarrollado fué entregado en su totalidad para que el personal de la D.G.O.S.G. lo utilizará en el mes de abril de 1989, para poderlo entregar se hizo una exposición de las partes que conforman este "software", se explicó detalladamente los pasos a seguir y el llenado de las pantallas, así como el tipo de información que debía ser capturada en los distintos campos y los tipos de errores que nosotros pensamos serían los más frecuentes en el usuario. Se utilizaron todas las opciones que nos da el sistema y se mostraron los resultados al usuario, quedando este conforme con los mismos. Para poder liberar el sistema se aplicaron las pruebas que se explican a continuación.

#### 4.3.2 Probando el "software" del Sistema de Administración Costo de Obra

##### 4.3.2.1 Pasos en las pruebas del "software" del Sistema de Administración Costo de Obra

Para probar el "software" como lo indica el capítulo de metodología, dividiremos las pruebas en cuatro partes, la primera será la prueba de unidad, la segunda será la prueba de integración, la tercera será la prueba de validación y por último probaremos el sistema completo.

##### 4.3.2.1.1 Pruebas de unidad del Sistema de Administración Costo de Obra

Estas pruebas se enfocan a cada uno de los módulos que forman el sistema, en nuestros módulos se probó lo siguiente.

- La estructura de datos local. Todas las variables utilizadas y los registros fueron verificados para que no se presentaran deficiencias en los mismos.
- Las condiciones límite. Existen bastantes condiciones dentro de nuestro "software" y cada una fué probada hasta llegar al punto límite para asegurar así su eficiencia y confiabilidad.
- Las interfaces del módulo. Estas pruebas se detallan dentro de las pruebas de integración.

Enseguida, respondiendo a las preguntas planteadas por el autor Myers (7) para el manejo de archivos y que nosotros consideramos necesario hacerlos, tendremos las siguientes respuestas.

1. ¿Son correctos los atributos de los archivos? SI
2. ¿Están correctamente escritos los sentencias open? SI
3. ¿El tamaño del buffer es acorde con el tamaño del registro? SI
4. ¿Se abrieron los archivos antes de usarse? SI
5. ¿Se maneja la condición de fin de archivo? SI, junto con las demás condiciones de error.

En cuanto a los errores que podían ocurrir como declaraciones inconsistentes, valores asignados incorrectos, nombres de variables incorrectos, etc., todos estos fueron corregidos en el momento que se presentaron.

#### 4.2.1.2 Pruebas de integración del Sistema de Administración Costo de Obra

Aquí probamos todo lo referente a las ligas que existen entre los diferentes módulos; para realizar estas pruebas, contestamos a las preguntas planteadas en el capítulo de metodología de implementación.

- ¿El número de parámetros de entrada del módulo llamado es igual al número de parámetros de salida del módulo que llama? SI, los números de parámetros son los mismos.
- ¿Los tipos de los parámetros son los mismos? SI, se revisó cada una de las parejas para que no existieran problemas con los tipos.
- ¿El orden de los parámetros es el mismo en ambos módulos? SI, el orden fue revisado por parejas.
- ¿Son respetadas las variables globales dentro del módulo? SI, se tuvo bastante cuidado con las variables globales.

Todas las conexiones entre los diferentes módulos fueron verificadas una a una respondiendo a las preguntas planteadas anteriormente.

#### 4.3.2.1.3 Pruebas de validación del Sistema de Administración Costo de Obra

Para realizar esta prueba, se citó al usuario en la D.G.O.S.G., para mostrarle los resultados que entregaba el sistema y la forma de manejo. Una vez mostrado el sistema el usuario mostró sus inquietudes y estas fueron tomadas en cuenta para hacerle mejoras o cambios al sistema.

Una vez terminadas las mejoras se volvió a citar al usuario y en esta ocasión se llegó a la conclusión de que el sistema "SI" cumplía con los requerimientos planteados en un principio.

#### 4.3.2.1.4 Probando las funciones del Sistema de Administración Costo de Obra

Como ya se comentó anteriormente, el día que el sistema fue entregado, se probó en su totalidad frente al usuario quedando este satisfecho con los resultados. Tomamos este punto como la prueba aquí requerida del sistema completo.

#### 4.3.3 Mantenimiento del "software" del Sistema de Administración Costo de Obra

##### 4.3.3.1 Mantenimiento más fácil desde el desarrollo del Sistema de Administración Costo de Obra

Para poder tener un mantenimiento poco complicado, se tomaron en cuenta los siguientes puntos.

Durante el análisis se realizó un desarrollo estándar.

Durante el diseño se enfatizó la modularidad, la claridad y se utilizó una notación



estandar en los diagramas de flujos y en las cartas de estructuras de datos. Durante la implementación se manejó la programación estructurada, el código se intentó que fuera poco complicado, se usaron constantes simbólicas y cada listado está perfectamente documentado internamente.

#### **4.3.3.2 Aspectos administrativos del mantenimiento del Sistema de Administración Costo de Obra**

Para poder dar un buen manejo a los requerimientos de mantenimiento, se recomienda tener una tarjeta en la cual se incluyan todos estos requerimientos.

Dentro de nuestra tarjeta de mantenimiento del sistema control costo de obra, tenemos anotados hasta la fecha dos requerimientos.

El primero fué cambiar los encabezados del reporte de la explosión de insumos, este requerimiento fué aceptado y se realizaron los cambios necesarios en aproximadamente cinco horas.

El segundo requerimiento era dentro del módulo de cambios de duración de la ruta crítica. Solicitaron que se pudiera cambiar la duración además del número de cuadrillas. Este requerimiento no fué aceptado puesto que si el usuario puede cambiar la duración el número de cuadrillas podría resultar un número con decimales y esto no es posible en términos reales, puesto que no se pueden tener trabajando 1.3 ó 6.8 cuadrillas (1.3 y 6.8 son ejemplos ficticios).

#### **4.3.3 Problemas durante el mantenimiento del Sistema de Administración Costo de Obra**

Durante el mantenimiento de cualquier sistema, las personas que lo desarrollaron se enfrentan a problemas como los siguientes:

- El usuario llega a solicitar cosas que no pueden ser. Un ejemplo de esto es el segundo requerimiento planteado anteriormente.
- El usuario comienza a solicitar cosas que nunca antes había planteado.
- El usuario no maneja bien la información y se queja del sistema, solicitando sea corregido.

Los anteriores y muchos otros son problemas que se deben tratar entre las personas que desarrollaron el "software" y el usuario para llegar a un buen término.

## 5. CONCLUSIONES

El presente trabajo sirvió para poder reafirmar los conocimientos adquiridos en cuanto al manejo de los sistemas computacionales; este manejo lo dividimos en tres etapas que son: la de análisis, diseño e implementación dentro del análisis planteamos la realización de un estudio de factibilidad el cual debe de presentarse como su nombre lo dice, la factibilidad que tiene el sistema a ser desarrollado, los puntos que nosotros proponemos a seguir dentro del estudio de factibilidad simplemente se enumeran y no se detallan pues existen gran variedad de formas para desarrollarlos y todas son validas.

La parte final de la metodología del análisis, toda la metodología del diseño y toda la metodología de la implementación están expuestas a detalle debido a que este es nuestra "TESIS" de como se debe llevar a cabo el desarrollo de un buen sistema computacional, es a mi parecer estas metodologías una buena ayuda para cualquier persona que quiera realizar el desarrollo de un sistema computacional, herramienta que no se encontró a mi gusto en ningún libro que heble sobre metodologías del "software".

Considero que el desarrollo de estas metodologías es parte importante dentro del presente trabajo.

Refiriendonos a la segunda parte importante dentro de este trabajo ubicamos el desarrollo del sistema de Administración Costo de Obra durante este desarrollo se aplicó la metodología que nosotros planteamos anteriormente y al mismo tiempo esta metodología fué sufriendo modificaciones que la hacían cada vez más práctica; el desarrollo del sistema nos permitió adquirir practica en aspectos como los siguientes:

- toma de notas en entrevistas con el usuario,
- toma de decisiones en el momento de presentarse los problemas,
- experiencia en el análisis diseño e implementación de sistemas,
- experiencia en el manejo de la información (normalización de archivos)
- conocimiento del equipo E-25 de Burroughs,
- otros.

En cuanto al resultado que arroja el presente trabajo, considero que realmente es una herramienta que dará ayuda al usuario y facilitará el manejo de su información, disminuirá el trabajo del personal de la D.G.O.S.G. teniendo así un poco mas de tiempo para el desarrollo de sus actividades.

En general el presente trabajo lo considero un buen final dentro de mi formación profesional en la Facultad de Ingeniería.

Atte.

Javier Velázquez Tzotza

## **BIBLIOGRAFIA**

1. Suárez Salazar, 1980  
"Costos y tiempos en edificación"  
Editorial Limusa  
3a. Edición.
  
2. Origel Couigno, 1988  
Apuntes de Ingeniería de programación  
Facultad de Ingeniería  
Apuntes no editados.
  
3. Softek  
Apuntes sobre desarrollo de "software"  
Empresa Softek.
  
4. Jean Martín  
"Computer Data-Base Organization"
  
5. Yourdon Edward  
"Estructures design: Fundamentals of a  
discipline of computer programs and  
system desing"  
Prinlece Hall
  
6. Pressma Roger, 1982  
"Software engineering: a practitioner's  
approach"  
McGrawHill
  
7. Myers, Glenford  
"The art of the software testing"  
"El arte de probar el software"  
Buenos Aires:  
Editorial Ateneo.