



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

FACULTAD DE INGENIERÍA

**“SISTEMA DE AUTOMATIZACIÓN DEL PROCESO
DE AUDITORÍA EN DEPENDENCIAS DEL
GOBIERNO FEDERAL”**

T E S I S

PARA OBTENER EL TÍTULO DE:

INGENIERO EN COMPUTACIÓN

P R E S E N T A:

JOSÉ ANTONIO CEJUDO IBARRA

DIRECTOR: ING. ROSA LOBERA SÁNCHEZ



MÉXICO, D.F.

2004

ÍNDICE GENERAL

1.- INTRODUCCIÓN.....	1
1.1 Objetivo.....	2
1.2 Fundamentos Teóricos.....	2
1.2.1 Metodologías de Desarrollo de Sistemas.....	2
1.2.1.1 Modelo Clásico.....	2
1.2.1.2 Modelo en Espiral.....	3
1.2.1.3 Modelo Orientado a Objetos.....	4
1.2.2 Sistemas Operativos.....	6
1.2.3 Lenguajes de Programación.....	7
1.2.3.1 Clases y Características.....	7
1.2.3.2 Generaciones de Lenguajes de Programación.....	8
1.2.3.3. Evolución de los Estilos de Programación.....	9
1.2.3.4. Herramientas de Programación.....	10
1.2.4 Bases de Datos.....	11
1.2.4.1 Definición.....	11
1.2.4.2 Elementos de un Sistema de Bases de Datos.....	12
1.2.4.3 Independencia de Datos.....	13
1.2.4.4 Integridad de los Datos.....	13
1.2.4.5 Modelos de Bases de Datos.....	14
1.2.4.5.1 Modelo Jerárquico.....	14
1.2.4.5.2 Modelo de Red.....	16
1.2.4.5.3 Modelo Relacional.....	17
1.2.5 Redes de Computadoras.....	19
1.2.5.1 Tipos.....	19
1.2.5.2 Equipos de Red.....	20
1.2.5.3 Modelo de Interconexión de Sistemas Abiertos Osi.....	21
1.2.5.4 Arquitectura Cliente/Servidor.....	24
2.- INGENIERÍA DE PROGRAMACIÓN.....	25
2.1 Estudio Inicial.....	26
2.1.1 Identificación de Problemas, Oportunidades Y Objetivos.....	26
2.1.2 Determinación de los Requerimientos de Información.....	26
2.1.3 Determinación de la Factibilidad.....	27
2.2 Análisis.....	28
2.3 Diseño.....	28
2.4 Desarrollo.....	29
2.5 Documentación.....	29
2.6 Pruebas e Implantación.....	30

2.6 Pruebas e Implantación	30
2.7 Evaluación y Mantenimiento.....	30
3.- ESTUDIO INICIAL.....	31
3.1 Identificación del Problema, Oportunidades y Objetivos	32
3.1.1 Descripción del Sistema Actual.....	32
3.1.1.1 Elaboración del Paca.....	32
3.1.1.2 Inicio de Auditoría.....	32
3.1.1.3 Planeación de Auditoría	33
3.1.1.4 Ejecución de Auditoría.....	33
3.1.1.5 Informe de Auditoría.....	33
3.1.1.6 Canalización de Informes	34
3.1.1.7 Reportes	34
3.1.1.8 Observaciones.....	34
3.1.2 Identificación del Problema	37
3.1.3 Objetivos.....	37
3.2 Análisis de Factibilidad.....	38
3.2.1 Factibilidad Técnica.....	38
3.2.2 Factibilidad Operativa.....	40
3.2.3 Factibilidad Económica	40
3.3 Análisis de Requerimientos de la Información	41
3.3.1 Entrevistas	41
3.3.2 Definición del Sistema y Módulos a Desarrollar.....	56
3.3.2.1 Módulo Programa Anual de Control y Auditoría (PACA).....	56
3.3.2.2 Inicio de Auditoría.....	57
3.3.2.2.1 Módulo Orden de Auditoría	57
3.3.2.2.2 Módulo de Consulta y Selección de Observaciones(Inventario)	57
3.3.2.2.3 Módulo Acta de Inicio.....	58
3.3.2.2.4 Módulo Carta Planeación.....	58
3.3.2.2.5 Módulo Cronograma de Actividades.....	58
3.3.2.3 Ejecución de Auditoría.....	58
3.3.2.3.1 Módulo Marco Conceptual	58
3.3.2.3.2 Módulo Cédulas de Observaciones	59
3.3.2.3.3 Módulo Cédulas de Seguimientos	59
3.3.2.4 Informe de Auditoría	60
3.3.2.4.1 Módulo Informe de Auditoría.....	60
3.3.2.4.2 Módulo Cédula Única de Auditoría	60
3.3.2.4.3 Módulo Evaluación de Auditoría.....	60
3.3.2.5 Módulo Catálogos.....	60
3.3.2.6 Módulo Reportes	61

4.- ANÁLISIS Y DISEÑO.....	63
4.1 Análisis	64
4.1.1 Análisis de Requerimientos del Sistema.....	64
4.1.2 Planeación de Actividades.....	66
4.1.3 Diagrama de Flujo de Datos (Dfd).....	68
4.1.4 Construcción del Diccionario de Datos	77
4.1.4.1 Definición	77
4.1.4.2 Construcción.....	77
4.1.4.2.1 Procesos.....	78
4.1.4.2.2 Flujo de Datos	79
4.1.4.2.3 Almacenes de Datos.....	80
4.1.4.2.4 Estructuras de Datos.....	81
4.2 Diseño.....	87
4.2.1 Diseño de Carta Estructurada.....	87
4.2.1.1 Diseño Descendente de Sistemas	87
4.2.1.2 Desarrollo Modular del Software.....	87
4.2.1.3 Diseño de La Carta Estructurada	87
4.2.2 Diseño de Procedimientos de Captura	91
4.2.3 Diseño de La Base de Datos.....	92
4.2.3.1 Normalización	93
4.2.3.1.1 Primera Forma Normal (Fn1)	93
4.2.3.1.2 Segunda Forma Normal (Fn2)	94
4.2.3.1.3 Tercera Forma Normal (Fn3)	95
4.2.4 Diseño de Entradas e Interface (Pantallas)	97
4.2.5 Diseño de Salidas	109
5.- DESARROLLO, DOCUMENTACIÓN Y PRUEBAS.....	113
5.1 Lineamientos del Desarrollo	114
5.1.1 Declaración de Constantes.....	115
5.1.2 Declaración de Variables	116
5.1.3 Declaración de Tipos.....	118
5.1.4 Declaración de Subrutinas	119
5.1.5 Declaración de Funciones	119
5.1.6 Comentarios.....	120
5.2 Desarrollo de Módulos.....	122
5.2.1 Módulos	122
5.2.1.1 Paca.....	122
5.2.1.2 Orden de Auditoría	123
5.2.1.3 Consulta del Inventario de Observaciones	125
5.2.1.4 Acta de Inicio.....	126

5.2.1.5 Carta Planeación	127
5.2.1.6 Cronograma de Actividades	128
5.2.1.7 Marco Conceptual.....	128
5.2.1.8 Cédula de Observaciones	130
5.2.1.9 Cédula de Seguimientos	131
5.2.1.10 Informe de Auditoría y Cédula Única de Auditoría	133
5.2.1.11 Evaluación de Auditoría.....	134
5.2.1.12 Catálogos.....	134
5.2.1.13 Reportes	135
5.2.2 Supervisión.....	136
5.3 Documentación.....	137
5.3.1 Manual de Usuario.....	137
5.3.2 Manual Técnico	140
5.4 Pruebas.....	141
5.4.1 Pruebas de Validación.....	141
5.4.2 Pruebas de Unidad.....	142
5.4.3 Pruebas de Integración o Integrales.....	142
6.- IMPLANTACIÓN, EVALUACIÓN Y MANTENIMIENTO....	145
6.1 Implantación	146
6.1.1 Verificación de Correcta Instalación	146
6.1.2 Capacitación de los Usuarios	146
6.1.3 Puesta en Operación.....	147
6.2 Evaluación.....	148
6.2.1 Técnicas de Evaluación	148
6.2.2 Evaluación del Sistema.....	148
6.3 Mantenimiento.....	150
6.3.1 Mantenimiento Preventivo.....	150
6.3.2 Mantenimiento Correctivo	150
6.3.3 Mantenimiento Perfectivo.....	151
CONCLUSIONES.....	153
APÉNDICES.....	157
BIBLIOGRAFÍA.....	163

CAPÍTULO 1



Introducción

En este capítulo se define el objetivo del desarrollo del “SAPA”, así como los fundamentos teóricos empleados para el desarrollo del sistema.

1.1 Objetivo

Implementar, a través de una metodología de desarrollo de sistemas, una aplicación que logre hacer más eficiente la operación en el área de la contraloría. El proceso principal que desempeña el área es el de realizar auditorías, apoyado en el almacenamiento y administración tradicional de la información (manejo de documentos en archiveros).

La implementación del sistema permitirá: un manejo más ágil, rápido y centralizado de la información (herramienta para una toma de decisiones veloz y precisa) y facilitará la supervisión de las auditorías generadas por los auditores del área. También evitará confusiones entre los auditores de la Contraloría Interna, pues se evitará la duplicidad de documentos y de datos.

1.2 Fundamentos Teóricos

1.2.1 Metodologías de Desarrollo de Sistemas

1.2.1.1 Modelo clásico

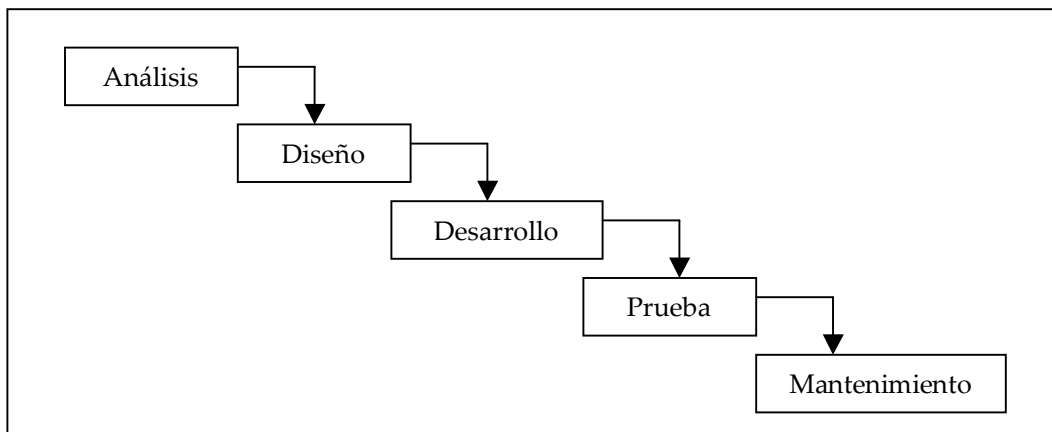


Figura 1.1 Diagrama de Modelo Clásico

La figura 1.1 ilustra el ciclo de vida clásico para la ingeniería del software. Consta en esencia de las siguientes etapas:

- **ANÁLISIS.-** Proceso de recopilación de información necesaria para el desarrollo del sistema. Para comprender la naturaleza de los programas que hay que construir; el ingeniero de software debe comprender el ámbito de la información del software así como la función, el rendimiento y las interfaces requeridas.
- **DISEÑO.-** Es un proceso multipaso enfocado en cuatro puntos distintos del programa: la estructura de los datos, la arquitectura del software, el detalle procedural y el diseño interfaz. El proceso de diseño traduce los requisitos en una representación del software que pueda ser establecida de forma que obtenga la calidad requerida antes de que comience la codificación. Al igual que los requisitos, el diseño se documenta y forma parte de la configuración del software.
- **DESARROLLO.-** El diseño debe de traducirse en una forma legible para la máquina. El paso de la codificación realiza esa tarea. Si el diseño se realiza de una manera detallada, la codificación puede realizarse de manera mecánica.
- **PRUEBA.-** Una vez generado el código, la prueba del programa comienza. La prueba se centra en la lógica interna del software, asegurando que todas las sentencias se han probado, y en las funciones externas, realizando pruebas que aseguren que la entrada definida produce los resultados deseados, utilizando datos o entradas reales o semejantes a éstos.
- **MANTENIMIENTO.-** La fase de mantenimiento se centra en: el cambio que va asociado a la corrección de errores, las adaptaciones requeridas por la evolución del entorno del software y las modificaciones debidas a los cambios de los requisitos del cliente dirigidos a ampliar o reforzar el sistema. La fase de mantenimiento vuelve a aplicar los pasos de las fases de definición y desarrollo, pero en el contexto del software ya existente. Durante la fase de mantenimiento se encuentran tres tipos de cambios:
 - a) *Corrección.-* Incluso al realizar las mejores actividades de garantía de calidad, es muy probable que el cliente descubra defectos en el software. El mantenimiento correctivo cambia el software para corregir los defectos.
 - b) *Adaptación.-* Con el paso del tiempo es probable que cambie el entorno original para el que se desarrolló el software. El mantenimiento adaptivo consiste en modificar el software para acomodarlo a los cambios de su entorno externo.
 - c) *Mejora.-* Conforme utiliza el software, el cliente/usuario puede descubrir funciones adicionales que podrían estar incorporadas al software. El mantenimiento perfectivo amplía el software más allá de sus requisitos funcionales originales.

El software sufrirá cambios después de que se libere, sólo que estos cambios serán los mínimos, en función de un mejor análisis y diseño.

1.2.1.2 Modelo en espiral

Utiliza un enfoque evolutivo para la ingeniería del software, permitiendo al desarrollador y al cliente entender y reaccionar a los riesgos en cada nivel evolutivo.

Utiliza la creación de prototipos como un mecanismo de reducción de riesgo, pero, lo que es más importante, permite a quien lo desarrolla aplicar el enfoque de creación en cualquier etapa de la evolución del producto. Mantiene el enfoque sistemático correspondiente a los pasos sugeridos por el ciclo de vida clásico, pero incorporándole dentro de un marco de trabajo iterativo que refleja de forma más objetiva el mundo real. El modelo en espiral demanda una consideración directa de los riesgos técnicos en todas las etapas del proyecto y, si se aplica adecuadamente, debe reducir los riesgos antes de que se conviertan en problemas.

El modelo en espiral para la ingeniería del software ha sido desarrollado para cubrir las mejores características tanto del ciclo de vida clásico, como de la creación de los prototipos, añadiendo al mismo tiempo un nuevo elemento: el análisis de riesgo. El modelo representado mediante la espiral de la figura 1.2 define cuatro actividades, representadas por los cuatro cuadrantes.

1. *Planificación*: determinación de objetivos, alternativas y restricciones.
2. *Análisis de riesgo*: análisis de alternativas e identificación/resolución de riesgos.
3. *Ingeniería*: desarrollo del producto de “siguiente nivel”.
4. *Evaluación del cliente*: valoración de los resultados de la reingeniería.

Un aspecto intrigante en el modelo en espiral se hace evidente cuando se considera la dimensión radial representada en la figura 1.2. Con cada iteración alrededor de la espiral (comenzando en el centro y siguiendo hacia el exterior), se construyen sucesivas versiones del software, cada vez más completas. Durante la primera vuelta alrededor de la espiral se definen los objetivos, las alternativas y las restricciones, y se analizan e identifican riesgos.

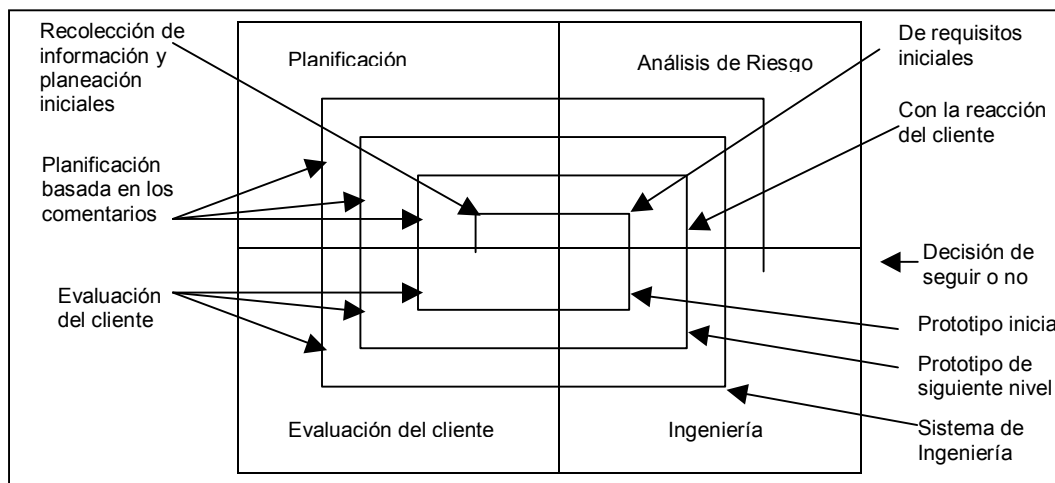


Figura 1.2 Diagrama del Modelo Espiral

Si el análisis de riesgo indica que hay una incertidumbre en los requisitos, se puede usar la creación de prototipos en el cuadrante de ingeniería para dar asistencia tanto al encargado como al cliente. Se pueden usar simulaciones y otros modelos para definir más el problema y refinar los requisitos.

El cliente evalúa el trabajo de ingeniería (cuadrante de evaluación del cliente) y sugiere modificaciones. Basándose en los comentarios del cliente se produce la siguiente fase de planificación y de análisis de riesgo. En cada bucle alrededor de la espiral, la culminación del análisis de riesgo resulta en una decisión de seguir o no. Si los riesgos son demasiado grandes; se puede dar por terminado el proyecto.

Sin embargo, en la mayoría de los casos se sigue avanzando alrededor del camino de la espiral, y ese camino lleva a los desarrolladores hacia fuera, hacia un modelo más completo del sistema, y finalmente, al propio sistema operacional. Cada vuelta alrededor de la espiral requiere ingeniería, que se puede llevar a cabo mediante el enfoque del ciclo de vida clásico o de la creación de prototipos. Debe tenerse en cuenta que el número de actividades de desarrollo que ocurren en el cuadrante inferior derecho aumenta al alejarse del centro de la espiral.

1.2.1.3 Modelo orientado a objetos

Es un proceso que posee una notación base para definir software, basado en el concepto de objetos que combinan estructura y comportamiento en una simple entidad. Describe y/o modela un sistema como un número de objetos relacionados que interactúan. Los objetos están organizados en clases. Este proceso no es necesariamente lineal pero sí predecible, repetible, evaluable y monitoreable.

La metodología de desarrollo orientada a objetos posee las siguientes tres partes:

- Proceso o actividades paso a paso empleado para construir software (USDP-Proceso unificado de desarrollo de software).
- Una notación o representación, a veces gráfica, de subsistemas que conforman un sistema y la interacción de los mismos subsistemas (UML).
- Un conjunto heurístico de reglas que describen como los sistemas deben de trabajar.

USDP es un proceso iterativo e incremental. Es iterativo porque consta de diferentes pasos a desarrollar y es incremental porque cada paso concluido representa un avance en el desarrollo del proyecto. Este proceso sugiere que, en lugar de resolver el problema en un paso, se lleven a cabo iteraciones a través del ciclo y se produzcan actualizaciones incrementales. Cada incremento mejora y corrige la anterior iteración.

Cada iteración debe realizar lo siguiente:

- Seleccionar y analizar los casos de uso relevantes.
- Crear y diseñar usando la arquitectura elegida.
- Implementar el diseño en componentes.
- Verificar que los componentes satisfagan los casos de uso.

Cuando se cumplen los objetivos de la iteración se procede con la siguiente iteración.

El USDP consta de cuatro fases principales:

- INICIO.- Se lanza el proyecto; se establece un razonamiento del negocio para el proyecto y las necesidades a resolver; se identifican los riesgos críticos y el alcance para una mejor comprensión del problema y, por último, se crean los documentos que explican las necesidades del negocio a resolver.
- ELABORACIÓN.- Se lleva a cabo la mayor parte de la tarea de análisis y diseño; se establece una arquitectura base; se monitorean los riesgos críticos y se crea un plan de construcción orientado a cubrir los objetivos del proyecto.
- CONSTRUCCIÓN.- Se lleva a cabo el desarrollo del proyecto de manera progresiva e incremental a través del desarrollo de diferentes módulos del proyecto.
- TRANSICIÓN.- Fase en la que se entrega la versión; se llevan a cabo las pruebas de desempeño; se realizan ajustes y se capacita al usuario.

A continuación se muestra la Tabla 1.1 con un diagrama de iteraciones, flujos de trabajo y fases contra tiempo:

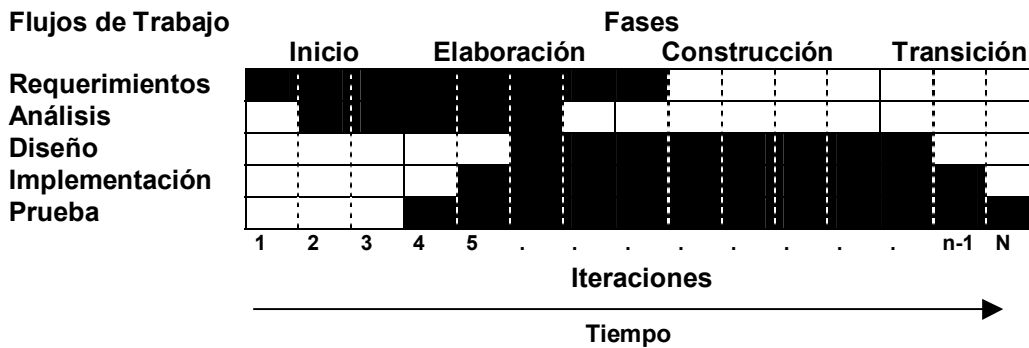


Tabla 1.1 diagrama de USDP

Cada iteración puede empezar después de (en algunos caso antes) la finalización de la iteración previa. Las iteraciones deben cumplir con su tiempo establecido y los problemas que surjan deben ser corregidos en posteriores iteraciones.

A veces se identifican problemas durante una iteración, que necesitan corrección en fases anteriores. En este caso se puede continuar y corregir en la siguiente iteración, agregar una

nueva iteración que solucione el problema o si el tiempo lo permite, regresar desde la iteración actual y corregir el problema.

1.2.2 Sistemas Operativos

Un “sistema operativo” (SO) es un programa que se carga en la memoria de la computadora por una rutina de inicio, y administra todos los demás programas, llamados aplicaciones. Las aplicaciones hacen uso del sistema operativo haciendo solicitudes de servicios a través de una o varias interfaces de programa de aplicación, conocidas también como APIs (por *Application Program Interface*). Además, los usuarios pueden interactuar directamente con el sistema operativo mediante una interfaz tal como un lenguaje de comandos.

Un sistema operativo realiza estos servicios para las aplicaciones:

- En sistemas operativos multitareas donde se pueden ejecutar múltiples programas al mismo tiempo, determina qué aplicaciones deben correr, en qué orden y cuánto tiempo se le debe brindar a cada aplicación antes de pasar el turno a otra.
- Administra el uso de la memoria interna entre múltiples aplicaciones.
- Maneja las entradas y salidas desde y hacia los dispositivos periféricos, tales como discos duros, impresoras y puertos de comunicación.
- Envía mensajes a las aplicaciones y al usuario acerca del estado de operación y si hubiera ocurrido algún error.
- Puede tomar el control de los llamados procesos en lote (por ejemplo, la impresión) de tal forma que la aplicación que lo generó se libere de este trabajo.
- En las computadoras que cuentan con procesamiento en paralelo, un sistema operativo puede "decidir" cómo dividir el programa de tal forma que se ejecute en más de un procesador a la vez.

Tipos de Sistemas Operativos

SISTEMAS POR LOTES (Mediados de los 50's - Mediados de los 60's).- Un sistema por lotes era aquel en el que las tareas eran agrupadas con las instrucciones necesarias para poder ser procesadas sin intervención externa.

SISTEMAS DE OPERACIÓN SIMULTÁNEA DE PERIFÉRICOS EN LÍNEA EN PROCESAMIENTO POR LOTES (Mediados de los 60's-finales de los 70's).- Este tipo de sistemas fueron los primeros y los más sencillos de los sistemas de multiprogramación. Una ventaja que poseían era que los resultados de cada una de las tareas estaban disponibles tan pronto como se completaban éstas de forma individual en lugar de esperar a que se completara todo el ciclo de tareas en proceso.

SISTEMAS DE MULTIPROGRAMACIÓN (multiprocesamiento) (60's-presente).- Los sistemas de multiprocesamiento requieren una cantidad de memoria tal, que puedan cargar varias tareas a

la vez y ciclar a través de ellas siguiendo un orden determinado, trabajando en cada una por un período específico de tiempo.

SISTEMA DE TIEMPO COMPARTIDO (70's-presente). Este tipo de sistemas surgió como respuesta a la necesidad de los usuarios de computadoras de tener acceso total los recursos de la máquina sin tener que hacerlo a través de un proceso de lotes. Existen muchos tipos de sistemas de tiempo compartido. Un tipo es representado por computadoras VAX/VMS y las estaciones de trabajo de UNIX. En estas computadoras se tienen procesos completos en memoria y éstos son ejecutados parcialmente brindando períodos de tiempo de procesamiento a cada uno.

Algunos, ejemplos de sistemas operativos que se utilizan en la actualidad son *Windows 9x/NT/2000*, *UNIX (Hp-ux, Linux, Solaris)*, *open VMS*, entre otros.

1.2.3 Lenguajes de Programación

1.2.3.1 Clases y características

Los “lenguajes de programación” son las herramientas que emplea el ingeniero de software para desarrollar las aplicaciones. A partir de los años sesenta han surgido cientos de lenguajes de programación. Muchos de ellos se han desarrollado para cumplir con objetivos particulares, otros han alcanzado un uso generalizado. Es importante para el ingeniero de software conocer las características generales que poseen los lenguajes, para poder tomar decisiones en cuanto a cuál utilizar para desarrollar un proyecto dado.

En ocasiones es necesario emplear más de un lenguaje en el desarrollo de un sistema para alcanzar el nivel de eficiencia requerido. Cuando el proyecto es demasiado grande, es válido dividirlo en varios módulos autónomos cooperativos, desarrollados con diferentes lenguajes para cubrir las responsabilidades de cada uno y tomando en cuenta, desde luego, el hecho de que deberá existir la manera de ligarlos entre sí de la forma más sencilla posible.

La decisión de qué lenguaje utilizar no es algo trivial pues de esto depende en gran parte el adecuado desarrollo del sistema. Al elegir una buena herramienta se reducen las dificultades de codificar un diseño, la cantidad de pruebas se minimiza y es más fácil darle mantenimiento. Dentro de las características que debe cumplir un lenguaje para desarrollo de sistemas destaca que debe tener facilidades de estructuración que permitan producir un código legible y lo más simplificado posible, tales como procedimientos y funciones globales; además, el lenguaje debe poder detectar errores en la compilación debidos a malas asignaciones de valores. Éstos son sólo algunos de los requisitos que debe cumplir un buen lenguaje de programación de sistemas aunque también se pueden considerar otros aspectos.

En ocasiones la elección del lenguaje a utilizar no depende completamente del desarrollador, sino del cliente, ya que éste puede decidir que el mantenimiento del sistema quede a cargo de su personal, por lo que deberá escribirse en un lenguaje que ellos conozcan.

El aspecto de la portabilidad de los lenguajes se debe considerar en caso de que el sistema esté orientado a trabajar sobre diversas configuraciones de hardware y sistemas operativos, ya que de ello puede depender la forma en que se efectúen algunos procesos tales como la impresión de reportes o la adquisición de datos, por citar sólo algunos casos.

El tamaño del proyecto puede en ocasiones limitar el uso de ciertos lenguajes; si es muy grande, es posible que incluso resulte necesario diseñar un lenguaje para desarrollarlo.

El conocimiento que poseen los miembros del equipo de desarrollo sobre un lenguaje, es también un punto a considerar, y es recomendable elegir un lenguaje conocido, siempre y cuando los aspectos mencionados previamente sean cubiertos. Por último, también hay que considerar el tipo de aplicación que se requiere programar.

1.2.3.2 Generaciones de lenguajes de programación

PRIMERA GENERACIÓN.- Se conocen como lenguajes máquina. Se componen de cadenas de 0 y 1's los cuales representan tanto las instrucciones como los datos con los cuales ha de actuar directamente el procesador. De esta manera, los lenguajes de Primera generación constituyen un conjunto de instrucciones orientadas a procesadores específicos.

SEGUNDA GENERACIÓN.- Esta generación abarca a los lenguajes ensambladores, los cuales, al igual que los lenguajes de primera generación, constituyen conjuntos de instrucciones para procesadores, con la diferencia de que no se representan como cadenas de dígitos binarios, sino como nombres cortos representativos llamados mnemónicos. Para ejecutar los programas escritos en este tipo de lenguajes se requiere de una herramienta llamada precisamente ensamblador, la cual convierte el lenguaje ensamblador a lenguaje máquina.

TERCERA GENERACIÓN.- Son considerados de "alto nivel", tales como *C* o *Java*. Su sintaxis contiene instrucciones cuyos nombres tienen relación con lo que realizan, lo cual facilita su aprendizaje. Un compilador convierte los enunciados de un lenguaje específico de alto nivel a lenguaje máquina.

CUARTA GENERACIÓN.- Se acercan más al lenguaje natural que los de tercera generación. Los lenguajes para acceder a bases de datos, como por ejemplo *SQL*, se definen normalmente dentro de esta generación. También se consideran de cuarta generación aquellos que utilizan una interfaz gráfica o visual de desarrollo para crear código fuente, el cual se compila casi siempre con un compilador de tercera o cuarta generación.

Actualmente existen diversos lenguajes de programación entre los que destacan: *Delphi* (*Object Pascal*), *Visual Basic*, *C++*, *Perl* y *Java* entre otros muchos.

1.2.3.3. Evolución de los estilos de programación

La programación de aplicaciones ha cambiado drásticamente en los últimos años y consecuentemente los estilos de programación.

La llamada “Crisis del Software”, que consiste en el incremento exponencial en el tamaño y tiempo de desarrollo de las aplicaciones, en función de un crecimiento lineal de la complejidad de los problemas a resolver, propició el surgimiento de la ingeniería del software.

Para resolver los problemas generados por la “Crisis del Software”, la ingeniería de software busca que el código de las aplicaciones cumpla las metas de modificabilidad, eficiencia, confiabilidad y comprensibilidad.

La ingeniería de software posee siete principios que le permiten resolver los problemas generados por la “Crisis del Software”: modularidad, localización, abstracción, ocultamiento de información, posibilidad de confirmación, uniformidad y terminalidad.

El primer modelo de programación que intentó cumplir estos principios es la programación estructurada. De una programación secuencial se pasó a una programación estructurada con muy buenos resultados.

La programación estructurada se basa en un diseño de la aplicación de lo general a lo particular (*top down*), y en el teorema de estructura. El teorema de estructura se refiere a que todo programa puede ser diseñado utilizando únicamente las estructuras básicas: secuencial, alternativa y repetitiva.

Un programa, por largo y complejo que sea, se puede diseñar anidando convenientemente estas tres estructuras básicas. Esto ayudará a que los programas sean fácilmente leídos por cualquier persona, a que el número de errores en el proceso de programación sea bajo y a que puedan mantenerse fácilmente.

A partir de la programación estructurada, los esfuerzos se centraron en potenciar la modularidad y la reutilización del código, lo que condujo a la programación orientada a objetos. Pero que los estilos hayan cambiado no significa que el último elimine a los anteriores; al contrario, se apoya en ellos.

La programación orientada a objetos (POO) es una forma de programación que emplea objetos (entidades de software), definidos por una clase, que poseen estado y comportamiento y que están ligados, mediante mensajes, para la solución de problemas.

Algunos conceptos importantes para la comprensión de la programación de objetos son los siguientes:

- **ABSTRACCIÓN.-** Generalización y centralización de la visión de algún problema para su modelado. Por ejemplo, una clase modela la generalidad de un tipo de objeto.
- **ENCAPSULAMIENTO.-** Característica que permite a los objetos tener su estado (conjunto de atributos), y comportamiento (conjunto de métodos), empaquetados dentro de ellos mismos.
- **HERENCIA.-** Propiedad que permite compartir estado y comportamiento de manera automática entre clases y subclases de objetos.
- **POLIMORFISMO.-** Esta característica permite implementar múltiples formas de un mismo método, dependiendo cada una de ellas de la clase sobre la que se realice la implementación.

Y los elementos más importantes de este tipo de programación son:

- **CLASE.-** Definición genérica de un tipo de objeto, es decir, el mapa a partir del cual se crean objetos de un tipo.
- **OBJETO.-** Encapsulamiento genérico de atributos y métodos para manipularlos.
- **MENSAJES.-** Indicaciones que se dan a los objetos para que realicen diferentes tareas. Estos mensajes pueden venir de un usuario o de otro objeto.
- **ATRIBUTOS.-** Variables encapsuladas en un objeto que definen su estado.
- **MÉTODOS.-** Determinan como debe actuar un objeto al recibir un mensaje.

1.2.3.4. Herramientas de programación

Dentro de las herramientas que proporcionan apoyo para la programación se encuentran las herramientas CASE (*Computer Aided Software Engineering Tools*-Herramientas de Ingeniería de Software Asistida por Computadora), que es una herramienta de soporte automatizado. Una herramienta CASE es una “Tecnología de Ambiente Integrado” porque integran el trabajo que se realiza sobre un sistema desde el principio hasta el final del ciclo de desarrollo.

Una herramienta CASE posee diferentes herramientas integradas (no solamente de programación) que buscan facilitar la interacción de un grupo de desarrollo al permitir:

- Que la elaboración de diagramas sea un proceso dinámico e imperativo, más que un proceso tedioso debido a cambios difíciles de reflejar en los mismos.
- Mejora la comprensión de cómo se encuentran relacionadas cada una de las partes del ciclo de desarrollo de un sistema.
- Promueve el cumplimiento de la metodología de desarrollo.
- Provee una mejor visión del sistema.
- Facilita la elaboración de la documentación

Una herramienta CASE posee en general las siguientes herramientas:

- **HERRAMIENTAS DE PLANIFICACIÓN DE SISTEMAS DE GESTIÓN.**- Ayuda a comprender cómo se mueve la información entre las distintas unidades organizativas de la empresa para la cual se desarrollará el sistema.
- **HERRAMIENTAS DE GESTIÓN DE PROYECTOS.**- Sirve para hacer estimaciones útiles de esfuerzo, costo y duración de un proyecto; definir una estructura de partición del trabajo y hacer una planificación y seguimiento realista.
- **HERRAMIENTAS DE SOPORTE.**- Incluyen herramientas de documentación, herramientas para la gestión de redes (correo electrónico, boletines electrónicos, etc.) y software del sistema, herramientas de control de calidad y herramientas de gestión de bases de datos.
- **HERRAMIENTAS DE ANÁLISIS Y DISEÑO.**- Permiten al ingeniero de software crear un modelo del sistema que va a construir. El modelo contiene una representación: del sistema a construir, de los datos, los procesos, especificaciones de control, etc.
- **HERRAMIENTAS DE PROGRAMACIÓN.**- Se engloban compiladores, editores, y los depuradores que se usan con los lenguajes de programación convencionales. Entran en esta categoría los entornos de programación visuales pertenecientes a los lenguajes de cuarta generación, los generadores de aplicaciones y los lenguajes de consulta a bases de datos.

1.2.4 Bases de Datos

1.2.4.1 Definición

Una “Base de Datos” puede definirse como una colección de datos interrelacionados entre sí y almacenados en conjunto, sin redundancias innecesarias, la cual tiene como finalidad servir a una o más aplicaciones de la mejor manera posible.

La representación gráfica de una Base de Datos es:

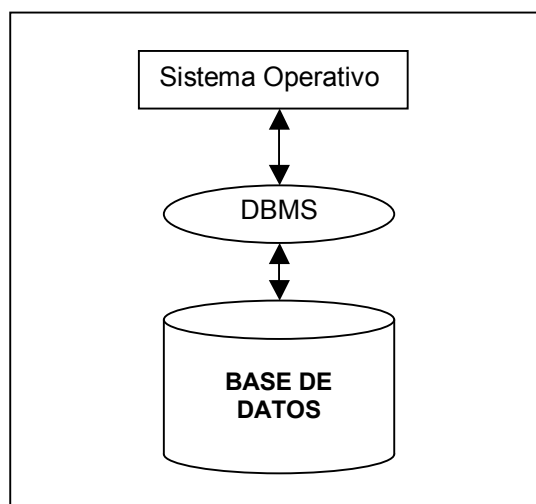


Figura 1.3 Base de datos

Los problemas de mantener grandes sistemas basados en archivos condujeron, hacia finales de los años sesenta, al desarrollo de los sistemas de gestión de Bases de Datos. La idea detrás de estos sistemas era sencilla, tomar la definición de los contenidos de un archivo y la estructura de programas individuales, y almacenarla junto con los datos en una Base de Datos.

Los sistemas de gestión de Bases de Datos organizan y estructuran los datos de tal modo que puedan ser recuperados y manipulados por usuarios y programas de aplicación. Las estructuras de los datos y las técnicas de acceso proporcionadas por un DBMS (*Data Base Management System*—Sistema Manejador de Bases de Datos), se denominan Modelo de Datos.

Utilizando la información de la Base de Datos, el DBMS que la controla pudo tomar un papel mucho más activo en la gestión de los datos y en los cambios a la estructura de la Base de Datos.

Cuando la gestión de Bases de Datos se popularizó durante los años setenta y ochenta, emergieron un puñado de modelos de datos populares. Cada uno de estos primeros modelos de datos tenía ventajas y desventajas que jugaron papeles importantes en el desarrollo del modelo de datos relacional. En muchos sentidos el modelo de datos relacional representó un intento de simplificar los modelos de datos anteriores.

1.2.4.2 Elementos de un Sistema de Bases de Datos

Un Sistema de Bases de Datos incluye cuatro elementos:

- **DATOS.**- Un Dato es una característica de un hecho o ente que por sí solo no es significativo. Sólo cuando están en conjunto adquieren su completo significado, es decir, se convierte en información.
- **HARDWARE.**- El Hardware de una Base de Datos. Consiste básicamente en el medio en que reside de manera física.
- **SOFTWARE.**- Entre las bases de Datos físicas y los usuarios se encuentra el sistema manejador de Base de Datos (*Data Base Management System*, DBMS), que es un conjunto de programas que crean las Bases de Datos, permiten el acceso a los registros (seguridad) y administran la información de acuerdo a los requerimientos de los usuarios.

Para realizar las funciones mencionadas, un DBMS tiene como mínimo los siguientes componentes:

- Un lenguaje de definición de datos (*Data Definition Lenguaje*, DDL) que permite crear la estructura de cada registro.
- Un Lenguaje de Manejo de Datos (*Data Manipulation Lenguaje*, DML) que permite agregar, borrar o modificar registros de la Base de Datos. Los lenguajes de consulta y los generadores de reportes forman parte del DML.
 - Un Diccionario de Datos (DD) que permite mantener la integridad de la Base de Datos, pues contiene datos de los datos, es decir, información referente a los formatos de los

registros, derechos de acceso, usuarios, etc. No debe confundirse este concepto con el Diccionario de Datos como herramienta de análisis de sistemas.

- Un Lenguaje de Programación para construir programas o reportes complejos. Actualmente algunos manejadores de Bases de Datos ofrecen generadores de aplicaciones como parte integral del mismo.

- USUARIOS.- Se considera que hay tres grandes tipos de usuarios:

- *Administrador de la Base de Datos.*- Es quien tiene a su cargo el diseño de la Base de Datos y el control de ellos. Entre otras funciones, el Administrador de la Base de Datos define la estructura y métodos de acceso a la Base de Datos, los tipos de autorización de acceso para los usuarios y los controles de integridad, además de establecer las estrategias de respaldo y recuperación de la información.

- *Programador de aplicaciones.*- Se encarga de escribir los programas que hacen uso de la Base de Datos.

- *Usuario final.*- Emplea la información de la Base de Datos por medio de una aplicación que facilite la manipulación de los datos.

1.2.4.3 Independencia de datos

Permite que los cambios en la organización física o estructura lógica de la base no afecten los programas de aplicación. Un sistema de base de datos ofrece dos tipos de Independencia de datos: física y lógica.

La independencia lógica se refiere a la protección de algunos programas de aplicación cuando se modifica el esquema (por ejemplo, Modelo de entidad relación). Por ejemplo, los campos en un archivo conceptual (por ejemplo, una tabla) en el esquema se pueden insertar o borrar, pero el programa de aplicación que usa ese archivo(tabla) debe continuar funcionando correctamente sin necesidad de modificaciones, siempre y cuando los campos insertados o eliminados no sean referenciados.

El DBMS logra esta independencia en dos pasos:

1. Las definiciones de datos para cada tipo de registro conceptual se describen en el esquema, el cual está separado físicamente de los programas de aplicación. Así, cualquier cambio en la definición de los datos resultará solamente en una modificación y recopilación del esquema, mientras que los programas de aplicación serán inmunes a los cambios.
2. Los registros lógicos externos requeridos por un programa de aplicación se definen en un subesquema (*Query*). El DBMS extrae de un esquema los elementos de datos necesarios del subesquema (*Query*). El Subesquema (*Query*) creado por el DBMS actúa como 'traductor intermedio' entre el usuario y los datos en cuestión. Si se cambian los datos de alguna forma, sólo se necesita cambiar el traductor intermedio para acomodar el cambio de datos.

La independencia física se refiere a la protección de los programas de aplicación contra los cambios en la estructura del sistema de archivos en donde se almacenan los archivos de datos, o cambios en los mismos nombres de los archivos de datos.

1.2.4.4 Integridad de los datos

El término “Integridad de los Datos” se refiere a medidas de seguridad usadas para mantener correctos los datos en la Base de Datos. Tal vez la mejor manera para explicar la integridad de los datos sea entendiendo la manera en que se introducen los errores en la Base de Datos. Las principales fuentes potenciales de error son:

- Falla del equipo (hardware).
- Falla del software (DBMS o aplicaciones que accesan los datos).
- Errores humanos.

Una falla del hardware, tal como un *head crash* (el choque de la cabeza de lectura/escritura sobre la superficie del disco), puede destruir físicamente los datos.

Otra causa de errores menos dramática son las fallas del sistema. Si se cae el sistema cuando se lleva a cabo una operación *WRITE*, esto puede resultar en una actualización incompleta causando inconsistencia en los datos. Los errores del software pueden deberse a "errores de lógica" (*bugs*), que son bastante familiares.

Para asegurar la integridad de los datos, no sólo son necesarios la confiabilidad física y lógica (hardware y software), sino también que los datos sean correctos.

La tecnología de los DBMS refuerza la integridad de los datos sin necesidad de demasiada programación extra por parte del usuario. La validación de datos propia del DBMS es un ejemplo de lo anterior.

1.2.4.5 Modelos de bases de datos

1.2.4.5.1 Modelo jerárquico

En un “Modelo jerárquico” los datos se estructuran de forma arborescente y las relaciones entre los diferentes tipos de registros se resuelven mediante punteros o enlaces entre ellos. Se establece una jerarquía de modo que las relaciones entre un registro y otro relacionado con él, son una relación “padre - hijo”, las cuales tiene como condición que un registro "hijo" no puede existir si no existe el registro “padre” asociado a él. La estructura es como la que se muestra en la figura 1.4

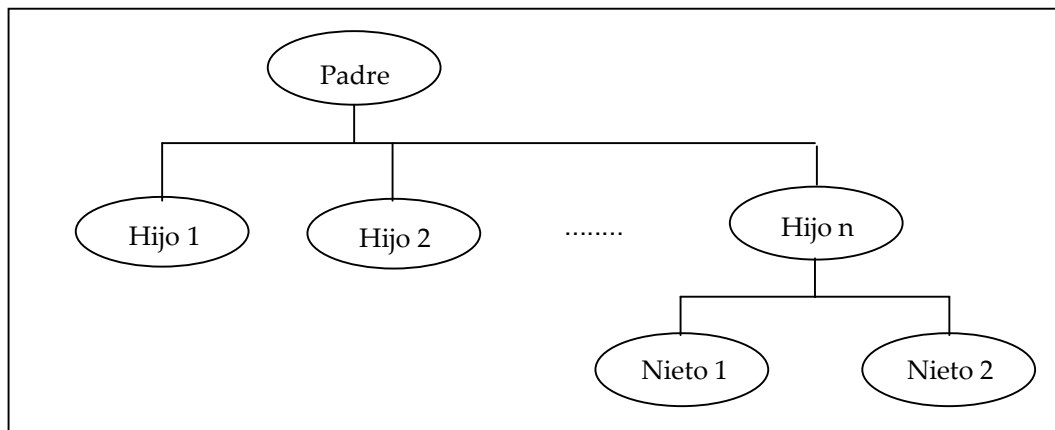


Figura 1.4 Modelo jerárquico

Con excepción de la raíz, todo nodo está relacionado con otro nodo de nivel más alto llamado padre. Ningún elemento puede tener más de un padre, sin embargo, todo elemento puede tener más elementos de nivel más bajo llamados hijos. Los elementos que se encuentran en las puntas de las ramas, es decir, que no tienen hijos, se les llama hojas.

Los árboles se utilizan tanto para las descripciones lógicas, donde describen las relaciones que existen entre los tipos de segmentos o tipos de registro, como para las descripciones físicas donde se le emplea para describir conjuntos de apuntadores y relaciones entre entradas en los índices. Dentro del modelo jerárquico existen diferentes tipos de jerarquías que son:

- *Jerarquías Balanceadas.*- En una jerarquía balanceada todos los nodos deben tener el mismo número de ramas; el árbol empieza por la raíz en forma descendente y va avanzando de izquierda a derecha.
- *Jerarquías Desbalanceadas.*- La mayoría de las organizaciones de datos por supuesto que no encajan en una estructura jerárquica balanceada, por lo que requieren números de ramas diferentes en los diversos nodos, llamadas jerarquías desbalanceadas. En este tipo de jerarquías el número de nodos puede variar respecto al número de ramas.
- *Jerarquías Binarias.*- La jerarquía binaria es una categoría especial de la estructura jerárquica balanceada, que admite sólo dos nodos por rama. Las jerarquías binarias al igual que otras jerarquías balanceadas son principalmente utilizadas en la representación física de los datos y no en las representaciones lógicas.
- *Jerarquías n-arias.*- Este tipo de jerarquía es problemática porque los vínculos entre las ramas y nodos son M:N (muchos a muchos). Este tipo de jerarquías utiliza tipos de registros virtuales, también llamados apuntadores.

La idea es incluir más de un esquema jerárquico en el esquema de la base de datos jerárquica y usar apuntadores de los nodos de un esquema jerárquico al otro para representar los vínculos.

Las estructuras jerárquicas implican, por lo general, que hay una correspondencia simple entre hijo-padre, es decir, cada hijo tiene sólo un padre; sin embargo, la correspondiente inversa es compleja, de uno a muchas, porque cada padre puede tener varios hijos.

1.2.4.5.2 Modelo de red

El “Modelo de red” se obtiene cuando se permiten relaciones entre "hermanos", es decir, entre registros de un mismo padre lógico. En ella se pueden construir esquemas como el que se muestra en la figura 1.5

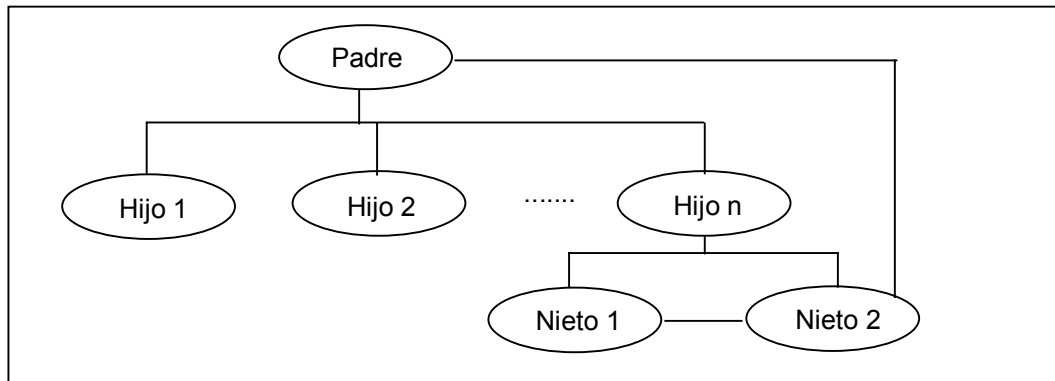


Figura 1.5 Modelo de red

Se debe considerar al modelo jerárquico como un tipo de modelo de datos en red con ciertas restricciones.

Una desventaja de este modelo es que algunas estructuras de red comienzan a tomar una apariencia de "telaraña", con apuntadores que salen en todas direcciones, por lo cual podrían llegar a ser complejas y difíciles de modificar.

Si en una relación entre datos un hijo tiene más de un padre, la relación no puede ser descrita por medio de un árbol o estructura jerárquica, sino como una estructura de red.

En un Modelo de Red cualquier elemento puede vincularse con cualquier otro elemento. Al igual que en la estructura jerárquica, la estructura de red puede ser descrita en términos de padres e hijos, y dibujada de cierta forma que los padres siempre van arriba de los hijos. Pero en la estructura en red, un hijo puede tener más de un padre.

En algunas estructuras de red es natural referirse a niveles, como ocurre con las estructuras jerárquicas.

En las estructuras de red que representan relaciones entre tipos de registros, la correspondencia entre padres e hijos es similar a la que existe en un árbol. Sin embargo, en un Modelo de Red es común que la estructura sea compleja, es decir, que los hijos tengan muchos padres, así como que los padres tengan muchos nodos hijos. Las estructuras complejas requieren métodos más elaborados para su representación física.

Las estructuras de red se pueden reducir a una forma más simple introduciéndoles algo de redundancia. En algunos casos la redundancia no es mucha y se tolera sin problemas.

Una estructura de red no es tan fácil de convertir en una estructura de árbol, ya que se requieren dos arboles para ser representada. Cada relación con correspondencia compleja en los dos sentidos tiene que ser remplazada por dos vinculaciones del tipo de estructura jerárquica.

1.2.4.5.3 Modelo Relacional

El Modelo Relacional de datos es definido en 1970 por el Dr. F. Codd, que lo define por medio de una serie de reglas cuyo objetivo es lograr la independencia de la representación lógica de los datos de su almacenamiento físico.

El Modelo Relacional es una estructura lógica en vez de física.

El modelo relacional de bases de datos consta de una estructura de datos, un conjunto de operadores relacionales y dos reglas de integridad. Estos tres componentes están vinculados con la definición, la manipulación y la integridad de los datos, respectivamente.

La estructura de datos es la denominada relación o tabla. El conjunto de operadores relacionales es lo que Codd definió como "Álgebra relacional". Tales operadores producen nuevas relaciones a partir de otras existentes. Las dos reglas de integridad son las de integridad de entidad y de integridad referencial.

El modelo relacional posee simplicidad en su estructura de datos. La manera más simple de definir una relación es como una tabla de datos. Una relación está formada por tuplas en la que cada tupla tiene elementos llamados atributos. Una tupla equivale a un renglón dentro de una tabla y un atributo a una columna.

El número de los atributos en una relación es el grado de la relación y el número de tuplas es la cardinalidad. El grado de una relación normalmente no cambia después de su creación, en cambio la cardinalidad varía según se agregan nuevas tuplas o se eliminan las existentes.

RFC	Nombre	Domicilio	Teléfono
GOPE760820	Enrique Posa Gómez	Mitla 72	515-12-83
ROLA781212	Alfredo Roca Luna	Monte Albán 124	755-33-80
LUMD821109	Diana Mendoza Pérez	Cuicuilco 18	373-45-90
OROI680404	Isabel Ortega Oca	Vértiz 23	553-93-75

Tabla 1.2 Relación o Tabla

En el ejemplo de la Tabla 1.2 tenemos los atributos RFC, Nombre, Domicilio y Teléfono, que al mismo tiempo nos indican que el grado de la relación es cuatro. También podemos ver que contamos con cuatro tuplas(renglones), por lo que la cardinalidad es cuatro.

El conjunto de valores de donde un atributo obtiene sus valores es el dominio del atributo. Los dominios son conceptuales en su naturaleza y pueden o no estar explícitamente contenidos en la base de datos. En el ejemplo anterior, si la tabla se refiriera (pues no se especifica), a los nombres de los empleados de una empresa, el dominio sería el total de los nombres de todos los empleados de la empresa.

Ya que en una tabla no está permitido tener renglones duplicados, cada tabla debe tener una llave primaria, es decir, un atributo o una combinación de éstos que identifique de manera única cada renglón dentro de la tabla.

Las llaves primarias son de fundamental importancia en el modelo relacional, ya que para localizar un renglón dado en una base de datos, es necesaria la combinación del nombre de la tabla y del valor de la llave primaria. En el ejemplo anterior la llave primaria es el RFC, pues es un valor diferente para cada empleado.

Toda relación o tabla debe cumplir las siguientes reglas:

- Toda relación o tabla debe tener un nombre único y distinto de otros dentro de la misma base de datos.
- Cada atributo o columna debe tener un nombre propio, único y distinto de otros, dentro de la misma tabla.
- Cada celda o intersección renglón-columna debe aceptar exactamente uno y sólo un valor y nunca un conjunto o arreglo de datos.
- Los valores en una misma columna deben ser del mismo tipo.
- Cada tupla (renglón) de la tabla debe ser distinto. No debe haber renglones idénticos.
- El orden de los renglones no es significativo.
- El orden de las columnas no es significativo.

El concepto de llave foránea es otro aspecto fundamental en las bases de datos relacionales. Una llave foránea es un atributo o una combinación de éstos, dentro de una relación R, que es la llave primaria de otra relación S.

Una base de datos relacional es un conjunto de tablas y las relaciones que existen entre ellas. Los conceptos del modelo relacional son aplicables únicamente a los niveles externo y conceptual de bases de datos. De hecho, el modelo relacional no propone nada en el ámbito interno, más bien se refiere a la manera en que los usuarios ven a la base de datos.

Las principales ventajas del modelo relacional son:

- *Simplicidad*: es decir, la facilidad de comprensión por parte del usuario final. El usuario final se enfrenta con un modelo de datos sencillo. Un modelo relacional es prácticamente lo que el usuario ve, y no necesariamente lo que se implantará físicamente.

- *Independencia de los datos:* esto debe constituir uno de los principales objetivos de cualquier modelo de datos. El modelo relacional elimina los detalles relativos a la estructura del almacenamiento. Este modelo proporciona un grado de independencia de los datos relativamente más alta que los dos modelos que se han discutido.
- *Fundamentos teóricos:* el modelo relacional está basado en la bien desarrollada teoría matemática de las relaciones. El riguroso método del diseño de una base de datos (usando la normalización) da a este modelo un fundamento sólido.
- *Búsquedas:* pueden ser mucho más rápidas que en los sistemas que deben seguir una cadena de apuntadores.
- *Operaciones de proyección:* son fáciles de implementar, la creación de nuevas relaciones necesarias para aplicaciones particulares resulta fácil de realizar.
- *Estructuras relacionales:* son mucho más fáciles de modificar que las estructuras jerárquicas o de red. En ambientes donde la flexibilidad es importante, esto es primordial.

Este modelo puede considerarse como una forma de representación universal, ya que es relativamente fácil convertir casi cualquier otro tipo de estructura o modelo de Bases de Datos al esquema relacional.

La representación tabular usada en el esquema relacional es fácil de comprender por los usuarios y fácil de implementar en el sistema físico de Bases de Datos.

El Diagrama Entidad-Relación o DER es actualmente utilizado como una herramienta para presentar los requerimientos de datos durante el diseño conceptual de Bases de Datos Relacional.

1.2.5 Redes de Computadoras

Se puede definir una “red de computadoras” como un conjunto de elementos conectados entre sí con la finalidad de compartir información y recursos.

1.2.5.1 Tipos

Las redes informáticas se pueden clasificar según su extensión o distribución geográfica:

- *Locales* o LAN (*Local Area Network*).- Una red de área local (LAN) es una red de "alta" velocidad (centenares de Megabits), generalmente confinada a un mismo piso o edificio.
- *Metropolitanas* o MAN (*Metropolitan Area Network*).- Una red de área metropolitana es un sistema de interconexión de equipos informáticos distribuidos en una zona que abarca diversos edificios, por medios pertenecientes a la misma organización propietaria de los equipos. Habitualmente, este tipo de redes se utiliza para interconectar redes de área local.

- *Extensas* o WAN (*Wide Area Network*).- Una red de área extensa es un sistema de interconexión de equipos informáticos geográficamente dispersos, incluso en continentes distintos. Las líneas utilizadas para realizar esta interconexión suelen ser parte de las redes públicas de transmisión de datos.

También se pueden clasificar por su Topología (forma lógica de una red):

- *Anillo*.- Es una de las tres principales topologías de red. Las estaciones están unidas una con otra, formando un círculo por medio de un cable común. Las señales circulan en un solo sentido alrededor del círculo, regenerándose en cada nodo.
- *Estrella*.- La red se une en un único punto, normalmente con control centralizado, como un concentrador de cableado.
- *Bus*.- Es la tercera de las topologías principales. Las estaciones están conectadas por un único segmento de cable. A diferencia del anillo, el bus es pasivo, no se produce regeneración de las señales en cada nodo.
- *Árbol*.- Consiste en conectar a un bus principal (tronco) varios buses secundarios (ramas). Permite establecer una jerarquía clasificando a las estaciones en grupos y niveles según al nodo al que estén conectadas. También realiza la distinción basándose en la distancia jerárquica al nodo central.
- *Trama*.- Esta estructura de red es típica de las WAN, pero también se puede utilizar en algunas aplicaciones de redes locales (LAN). Los nodos están conectados cada uno con todos los demás.

1.2.5.2 Equipos de Red

Un “equipo de red” es un dispositivo que interactúa con otros y que lleva a cabo una función dentro de una red. A continuación se describen los más comunes:

- **HUBS**.- Son equipos que permiten estructurar el cableado de las redes. La variedad de tipos y características de éstos es muy grande. En un principio eran sólo concentradores de cableado, sin embargo cada vez disponen de un mayor número de capacidades como aislamiento de tramos de red, conmutación de las salidas, gestión remota, entre otras.
- **ROUTERS**.- Son equipos de interconexión de redes que actúan en el ámbito de los protocolos de red. Permite utilizar varios sistemas de interconexión mejorando el rendimiento de la transmisión entre redes. Su funcionamiento es más lento que los bridges pero su capacidad es mayor. Permiten, incluso, enlazar dos redes basadas en un protocolo, por medio de otra que utilice protocolo diferente.
- **SERVIDORES DE TERMINALES E IMPRESORAS**.- Son equipos que permiten la conexión a la red de equipos periféricos tanto para la entrada como para la salida de datos. Estos dispositivos se ofrecen en la red como recursos compartidos. Así una terminal conectada a uno de estos dispositivos puede establecer sesiones contra varias computadoras

multiusuario disponibles en la red. Igualmente, cualquier sistema de la red puede imprimir en las impresoras conectadas a un servidor.

- NIC/MAU (Tarjeta de red).- *Network Interface Card* (Tarjeta de interfaz de red) o *Medium Access Unit* (utilidad de acceso al medio). Es el dispositivo que conecta la estación (computadora u otro equipo de red), con el medio físico. Se suele hablar de tarjetas en el caso de las computadoras, ya que la presentación suele ser como una tarjeta de ampliación de las mismas, diferente de la placa del CPU, aunque cada vez son más los equipos que disponen de una interfaz de red incorporada, principalmente *Ethernet*.
- PROTOCOLOS.- Un protocolo es una serie de reglas que utilizan dos equipos en red para comunicarse entre sí. Existen los siguientes tipos de protocolos:

-Protocolos de bajo nivel.- El protocolo de bajo nivel es la forma en que las señales se transmiten por el cable, transportando datos, información y los procedimientos de control de uso del medio por los diferentes nodos. Los protocolos de bajo nivel más utilizados son: *Ethernet*, *Token Ring*, *Token Bus*, *FDDI*, *CDDI*, *HDLC*, *Frame Relay*, *ATM*.

-Protocolos de red.- El protocolo de red determina el modo y organización de la información (tanto datos como controles) para su transmisión por el medio físico con el protocolo de bajo nivel. Los protocolos de red más comunes son: *IPX/SPX*, *DECnet*, *X.25*, *TCP/IP*, *AppleTalk*, *NETBEUI*.

1.2.5.3 Modelo de interconexión de sistemas abiertos OSI

Un estándar internacional para la arquitectura por capas de redes, es el modelo de referencia ISO (*International Standards Organization*) para la interconexión de sistemas abiertos OSI (*Open Systems Interconnections*) que se muestra en la Tabla 1.3

APLICACIÓN
PRESENTACIÓN
SESIÓN
TRANSPORTE
RED
ENLACE DE DATOS
FÍSICO

Tabla 1.3 Niveles del modelo OSI

A continuación se muestra cada uno de los niveles y las funciones que llevan a cabo:

- Nivel Físico(1)

- Se relaciona con interfaces mecánicas y eléctricas.
- Detecta colisión de datos.
- Especifica cables, conectores y otros componentes.
- Transmite información a través del canal de comunicación.
- Establece, mantiene y desconecta enlaces físicos.
- Incluye software de interfaces de comunicación.

- Nivel de Enlace de Datos(2)
 - Proporciona una transferencia de datos fiable.
 - Divide los datos (paquetes) en marcos (frames).
 - Añade bits para la detección/corrección de errores.
 - Controla el acceso y uso del canal.
 - Resuelve problemas de pérdida, daño y duplicación de frames.
 - Envía información.
 - Añade símbolos para indicar el principio y el final del mensaje.
 - Servicios de conexión o desconexión.

- Nivel de Red (3)
 - Establece, mantiene y finaliza la conexión.
 - Determina cómo enviar los paquetes.
 - Divide los mensajes enviados en paquetes y los reúne.
 - Realiza control de congestión y flujo de datos.
 - Reconoce las prioridades del mensaje.
 - Envía mensajes en el orden correspondiente.

- Nivel de Transporte (4)
 - Establece una sesión fiable de envío directo (detección de errores y recuperación), una vez que la conexión se ha establecido.
 - Fragmentación del mensaje en paquetes (si no se ha hecho en el nivel 3).
 - Transmisión simultánea de sesiones diferentes con un mismo origen y un mismo destino (multiplexing).
 - Crea distintas conexiones en la red.
 - Controla la calidad del servicio.
 - Disgrega y reúne los mensajes.
 - Controla el flujo de datos (si no se ha hecho en el nivel 3).

- Nivel de Sesión (5)
 - Establece y controla recursos de sistemas dependientes.
 - Establece y finaliza la conexión.
 - Servicio de contabilidad.
 - Interfaz del usuario con la red.
 - Identificación de usuarios.
 - Controla el diálogo, organiza y sincroniza.

- Nivel de Presentación (6)
 - Cantidad de datos, seguridad, compresión y conversión de código.

- Se asegura que los datos siguen el código estándar (ASCII).
- Tramita el paso de servicios desde el nivel de sesión al de aplicación.

- Nivel de Aplicación (7)
 - Comprueba el *login* y el *password*.
 - Establece la semántica para el intercambio de información.
 - Transfiere, accede y maneja archivos.
 - Transfiere mensajes y correo electrónico.
 - Transfiere y manipula el trabajo.
 - Servicio de directorio.
 - Manejo de sistemas.
 - Protocolos industriales.
 - Acceso y manejo de bases de datos.
 - Terminales virtuales.

1.2.5.4 Arquitectura cliente/servidor

El término alude a un procesamiento colaborativo de datos entre dos o más computadoras conectadas en una red. El cliente es el dispositivo que visualiza las pantallas, cuadros y texto, o reproduce el audio distribuido por el servidor. El cliente puede ser una computadora personal o un asistente digital personal, incluso una televisión inteligente que pueda comprender datos digitales. El servidor es el depositario del sistema, base de datos, video digital, audio, fotografías digitales y texto, y los distribuye bajo demanda; debe ser una máquina capaz de almacenar los datos y ejecutar todo el software que suministra a los clientes.

Una red puede tener un servidor que distribuya datos a múltiples clientes a la vez. Un cliente también podría tener múltiples servidores enviando datos simultáneamente.

En el entorno de computación actual, una computadora *Macintosh*, *Windows*, *UNIX* puede ser un cliente. Cualquiera de estas plataformas puede actuar como servidor e incluso puede actuar como cliente y servidor simultáneamente. Esta doble función es posible debido a las capacidades multitarea de los modernos sistemas operativos.

CAPÍTULO 2



Ingeniería de Programación

Para el desarrollo del Sistema de Automatización del Proceso de Auditoría, "SAPA", se tomó el enfoque sistemático que se encuentra en el "Ciclo Clásico de Desarrollo de Sistemas". El "Ciclo Clásico" es un enfoque por etapas de análisis y de diseño que propone que el desarrollo de los sistemas mejora cuando existe un ciclo específico de actividades del equipo que construye el sistema.

En este caso el ciclo se divide en siete etapas. Las etapas no se presentan de manera independiente, en lugar de ello, se realizan al mismo tiempo diversas actividades, y a veces llegan a repetirse. El ciclo clásico de desarrollo de los sistemas transcurre en etapas, con actividades que con el tiempo terminan poco a poco.

2.1 Estudio Inicial

2.1.1 Identificación de Problemas, Oportunidades y Objetivos

En esta fase se analizan las causas que dan origen al desarrollo de un sistema (problemas u oportunidades de mejora), y se determinan los objetivos que debe cumplir el sistema a desarrollar.

La primera etapa requiere un análisis objetivo de la forma en la que se trabaja en la empresa. Esto debe llevarse a cabo con los integrantes de la misma y con los analistas y líderes de proyecto que desarrollarán el sistema. La intención es conocer de manera general los procesos que el sistema podría automatizar, dando con esto solución a problemas existentes e implementando mejoras. Los empleados de la empresa pueden dar las mejores nociones sobre las eventuales problemáticas, y es tarea de los analistas y líderes generar propuestas sobre las posibles mejoras.

Las "oportunidades" son situaciones susceptibles de mejorar por medio de un sistema de información; con esto, una empresa puede lograr alcanzar un mayor grado de eficiencia en la operación de sus procesos administrativos, productivos y/o de servicios.

Para definir los "objetivos" que deberá alcanzar un sistema, lo primero es conocer los objetivos que persigue la empresa u organización que lo requiere, y con base en ellos determinar si un sistema informático puede contribuir a su logro y de qué forma.

2.1.2 Determinación de los Requerimientos de Información

Los "requerimientos de información" se obtienen a partir de los usuarios involucrados en las áreas o departamentos de una empresa en donde operará el sistema. Pueden utilizarse diversos instrumentos para obtener los requerimientos: el muestreo, el estudio de los datos y formas usadas por la organización, la entrevista, los cuestionarios, la observación de la

conducta de quién toma las decisiones, así como de su ambiente; y también el desarrollo de prototipos.

En esta etapa la interacción con el usuario es necesaria pues se busca conocer cuál es la información necesaria para que pueda realizar su trabajo, y, a través de esto, conocer los fines personales y diferenciarlos de los fines de la empresa.

Al mismo tiempo que se recopila la información necesaria para el desarrollo del sistema, se elabora su definición así como los módulos que se desarrollarán.

2.1.3 Determinación de la Factibilidad

La “factibilidad” del desarrollo de un sistema es el estudio de la posibilidad real de llevarlo a cabo según las condiciones actuales tanto de la empresa que requiere su desarrollo como de la empresa que lo desarrolla.

El estudio de factibilidad es una herramienta que sirve a la alta dirección para tomar la decisión de llevar a cabo el desarrollo o no. No implica un estudio muy profundo, sino más bien la recopilación de información clave para el desarrollo.

Factibilidad significa que el proyecto sugerido auxilia a la organización paralograr sus objetivos centrales; asimismo que sea posible con los actuales recursos de la organización en las tres áreas siguientes:

Factibilidad técnica

- Mejora del sistema actual.
- Disponibilidad de la tecnología que satisfaga las necesidades de desarrollo y operación.

Factibilidad económica

- Costo del tiempo de los empleados dedicados al análisis y desarrollo.
- Costo del estudio de sistemas.
- Costo estimado del equipo.
- Costo del desarrollo/adquisición del software.

Factibilidad operativa

- Que el sistema operará cuando se instale.
- Si el sistema será utilizado.

La forma más común de obtener la información para el análisis de factibilidad es por medio de entrevistas que normalmente se efectúan con los solicitantes del sistema y con los directivos responsables de la toma de decisiones.

Es importante elegir el problema correcto, pero los estudios de factibilidad no deben tomar demasiado tiempo.

2.2 Análisis

En esta etapa el primer paso es desarrollar la planeación de actividades para el desarrollo del sistema (análisis, diseño, etc.). La planeación implica la determinación y organización del tiempo y recursos necesarios para la construcción de un sistema que cumpla con los requerimientos y objetivos establecidos en el estudio inicial. Como herramienta para el desarrollo de la planeación puede emplearse el diagrama de Gantt.

A continuación, y como parte inicial de proceso de análisis, se emplean técnicas especiales que facilitan su realización, como el uso de los diagramas de flujo de datos, que son una técnica estructurada para representar en forma gráfica la entrada y salida de datos y los procesos de la información de la empresa.

A partir del diagrama de flujo de datos se desarrolla el diccionario de datos que contiene todos los elementos de datos que utiliza el sistema, así como sus especificaciones.

Es en esta fase donde se lleva acabo, de ser necesario, el análisis de decisiones por medio un método como el lenguaje estructurado, las tablas de decisiones o los árboles de decisiones. También se analizan las decisiones estructuradas por realizar, que son decisiones donde las condiciones, condiciones alternativas, acciones y reglas de acción podrán determinarse. Existen tres métodos para el análisis de las decisiones estructuradas: el lenguaje estructurado, las tablas de decisiones y los árboles de decisiones.

No todas las decisiones en las empresas son estructuradas, pero es importante que las comprenda el analista de sistemas:

- Las decisiones semiestructuradas son decisiones tomadas bajo riesgo y con frecuencia se apoyan en los sistemas para minimizar los riesgos. Para analizar las decisiones semiestructuradas se evalúa el grado de complejidad del problema y el número de criterios o variables consideradas.
- Las decisiones hechas con base en numerosos factores se denominan decisiones de criterio múltiple. Como ejemplo de técnicas para el análisis de decisiones de criterio múltiple se tiene el proceso de intercambio y la aplicación de métodos de ponderado.

En sistemas, existen múltiples soluciones para cada problema y el planteamiento de estas soluciones se relaciona directamente con la experiencia y formación profesional del analista.

2.3 Diseño

En esta etapa se usa la información recolectada con anterioridad y se elabora el diseño lógico del sistema. En primer lugar se construye la “Carta Estructurada”, que es el documento donde se encuentra la forma en la que se relacionan los diferentes módulos del sistema y la manera en la que se navegará a través de ellos.

A continuación se diseñan procedimientos de captura de datos, buscando que los datos que se inserten al sistema sean correctos.

La etapa del diseño también incluye el diseño de los archivos o la base de datos. Para la construcción de la base de datos o los archivos es necesario contar con un diccionario de datos (elaborado durante la fase de análisis). Un buen diseño de base de datos proporcionará un sencillo y rápido acceso a la información.

Teniendo el diseño de la base de datos se puede continuar con el diseño de las entradas o pantallas del sistema. Lo anterior se lleva a cabo por medio de documentos que describen la funcionalidad, diseño gráfico, controles empleados y campos de la base de datos que usa cada pantalla. También es aquí donde se especifican los menús y mensajes que mostrará cada pantalla.

Por último, se diseñan las salidas hacia el usuario de acuerdo con sus necesidades de información. El diseño de salidas se refiere al diseño de reportes en pantalla y reportes impresos. Los formatos de diseño de reportes son muy parecidos a los formatos de diseños de pantallas, sólo que se orientan más hacia el diseño gráfico y la descripción de los datos que contienen.

2.4 Desarrollo

En esta etapa se construye el software con el equipo de programadores por medio de la herramienta de programación elegida para el desarrollo. También se definen los lineamientos de desarrollo del sistema a nivel código, es decir, los estándares de programación y si la programación va a ser modular, estructurada, orientada a objetos, etc.

El líder de proyecto toma los diseños desarrollados anteriormente y basándose en la planeación realizada en la fase de análisis, elabora planes de trabajo más detallados. Los diseños de pantallas y módulos de código pueden incluir un pseudocódigo que facilite la codificación de los programadores, aunque esto no es requisito, ya que la elaboración repetida de pseudocódigo consume mucho tiempo. En general será suficiente con una buena narrativa de la funcionalidad. El uso del pseudocódigo se recomienda cuando la codificación de las pantallas es demasiado compleja; también para evitar confusiones en el programador.

2.5 Documentación

En esta fase se trabaja con los usuarios y desarrolladores del sistema para escribir la documentación del software. La documentación dice al usuario cómo trabajar con el software y qué hacer en caso de algún problema. En esencia, la documentación incluye el manual de usuario y el manual técnico.

El manual de usuario especifica cómo se operará el sistema en su conjunto y cómo funciona cada uno de los módulos que lo componen.

El manual técnico incluye:

- Diccionario de datos.
- Modelo de entidad relación.
- Documentación de *Stored procedures*.
- Documentación de módulos entre otras cosas.

2.6 Pruebas e Implantación

El sistema de información debe probarse antes de utilizarse. Lo más recomendable es detectar los problemas antes de la entrega del sistema. Existen tres tipos de pruebas: “*validación*”, “*unidad*” y de “*integración*” o “*integrales*”. El programador realiza las pruebas de validación por su cuenta, las otras se llevan a cabo en colaboración con el analista de sistemas y el usuario. Inicialmente, se hacen una serie de pruebas con datos de prueba, para identificar las posibles fallas del sistema; finalmente, se utilizarán los datos reales.

En esta etapa se ayuda a implantar el sistema que incluye la capacitación de los usuarios, la verificación de la correcta instalación y su puesta en operación. La responsabilidad de la capacitación del usuario pertenece a la empresa que produce el software, en especial los líderes de proyecto y los analistas. Se necesita planear la suave transición que trae consigo un cambio de sistemas.

2.7 Evaluación y Mantenimiento

La evaluación del sistema es, en esencia, el hecho de que el usuario realmente llegue a usar el sistema, satisfaciendo así sus necesidades de información. Cada una de las etapas de la construcción de un sistema tiene una parte de evaluación y la evaluación final es la que recoge todas las impresiones de la operación del sistema. Son los usuarios los que evalúan el desempeño del sistema.

El mantenimiento del sistema comienza en esta etapa; después se realizará de forma rutinaria a lo largo de toda la vida del sistema. Las actividades de mantenimiento son parte del trabajo de todos los programadores en las empresas por lo que lleva a consumir grandes cantidades de dinero en esta actividad; sin embargo, mientras mejor sea el análisis, diseño y desarrollo de un sistema, el costo de su mantenimiento será menor. Existen tres tipos de mantenimiento en esencia: “*preventivo*”, “*correctivo*” y “*perfectivo*”.

CAPÍTULO 3



Estudio Inicial

3.1 Identificación del Problema, Oportunidades y Objetivos

3.1.1 Descripción del Sistema Actual

Con base en las entrevistas iniciales hechas al usuario del sistema, se logró determinar de manera general el funcionamiento del proceso de elaboración de auditorías llevado a cabo por la contraloría interna.

La Contraloría Interna realiza una serie de auditorías periódicas para llevar un mejor control de la gestión de las diferentes áreas de cualquier dependencia de gobierno. Se utilizan una serie de formatos establecidos y otros definidos por ellos mismos con los que se recaba la información necesaria para la ejecución de las auditorías. Los documentos mencionados permiten la administración de las observaciones arrojadas por las auditorías para un mejor seguimiento.

El Departamento de Control y Evaluación captura en el Sistema de Información Periódica (SIP), la información requerida, que proviene de los informes resultantes de las auditorías practicadas. El cómo se realizan las actividades y procesos se menciona a continuación.

3.1.1.1 Elaboración del PACA

Durante el mes de noviembre de cada año, la Secretaría de Contraloría y Desarrollo Administrativo (SECODAM), solicita al Órgano Interno de Control de las Dependencias de Gobierno, el Programa Anual de Control y Auditorías (PACA), para tener conocimiento de las auditorías a realizar el próximo año dentro de la Institución de Gobierno, aunque en determinado momento se puede cancelar o agregar una auditoría previa autorización de la autoridad correspondiente.

3.1.1.2 Inicio de auditoría

La Dirección de Supervisión y Auditoría elabora el PACA, lo envía a la SECODAM para su registro y una vez obtenida su aprobación -en enero de cada año-, se inician las revisiones; posteriormente se designa al o los auditores que comenzarán por girar la orden de auditoría del área a auditar, que es donde se aclara, con base en fundamentos legales, la revisión que se llevará a cabo, el área especificada; asimismo se mencionan de manera general los alcances de la auditoría y el período por revisar, así como el tiempo en el que se practicará la revisión. Para el caso de las auditorías de seguimiento se mencionarán los antecedentes de las revisiones donde surgieron las observaciones sujetas a seguimiento, también se levanta el acta de inicio, documento donde se narran las circunstancias específicas del inicio de una auditoría. Contiene datos en detalle de actividades, documentos y personas que intervienen en el acto.

3.1.1.3 Planeación de auditoría

Dentro de la “planeación de la auditoría” se realiza la Carta Planeación donde se hace mención de los antecedentes del área programada o rubro por auditar, describiendo su estructura, marco legal aplicable, principales políticas y funciones, actividades que realiza, distribución física de las áreas de trabajo, resultados más importantes en auditorías anteriores, información financiera, presupuestaria y aquellos elementos que permitan al auditor tener un conocimiento general antes de iniciar la auditoría, así como la finalidad y alcances de la revisión. Se elabora el Cronograma de Actividades a desarrollar durante la auditoría, que se refiere a las acciones que de manera general se ejecutarán durante el tiempo programado de la revisión, y que son: inicio de la auditoría, levantamiento del acta, entrevistas con servidores públicos, solicitud de información, visitas físicas a instalaciones u oficinas, período de análisis del área, programa o rubro, comentario de observaciones y elaboración del informe, entre otros.

3.1.1.4 Ejecución de auditoría

Se ejecuta la auditoría de acuerdo con lo planeado, en donde se incluyen: el Marco Conceptual, que describe las cifras totales de las operaciones que se analizarán, tanto las financieras como las físicas. En todos los casos debe mencionarse la situación de los datos o cifras que se están manejando, los criterios utilizados para seleccionar la muestra, y, además, las técnicas específicas a desarrollar y los procedimientos que el auditor llevará a cabo para cumplir con su objetivo. El desglose de los procedimientos debe tener un orden cronológico y ser congruente con las operaciones a revisar. Se desagregarán tantos procedimientos como profundidad se dé al aspecto por revisar.

La aplicación de los procedimientos de Auditoría permite comprobar los hechos u operaciones, plasmando en cédulas de observaciones o seguimiento, según sea el caso, las desviaciones detectadas.

3.1.1.5 Informe de auditoría

Se realiza el Informe de Auditorías Integrales, Específicas, de Evaluación de Programa y de Desempeño o Seguimiento, según sea el caso, e incluyendo el nombre, sector y registro presupuestario de la Entidad de Gobierno. El informe es numerado con el número consecutivo que corresponda, el número total de hojas que lo integran; se señala el nombre del área auditada, el tipo y número de la auditoría de acuerdo con el programa anual de control y auditoría, así como los resultados de la revisión practicada.

En el expediente de las Auditorías se incluye la Cédula Única de Auditoría donde se lleva un control de los días/hombre utilizados en la realización de la auditoría, así como la cantidad de observaciones establecidas en la revisión; para las Auditorías Integrales deberá anotarse la cantidad de observaciones determinadas por cada rubro. En caso de una auditoría de seguimiento se anotará la cantidad de observaciones, de acuerdo con cada concepto, así como

la cantidad de observaciones a las que se les dio seguimiento, a aquellas a las que se les dio solución y las que están siendo atendidas.

3.1.1.6 Canalización de informes

Una vez que el informe ha sido realizado y revisado por el Director de Supervisión y Auditoría, se entrega al Contralor junto con los papeles de trabajo, que a su vez envía el informe a SECODAM con copia al Presidente de la Institución de Gobierno, al titular de la Unidad Administrativa auditada y otra junto con los papeles de trabajo al Departamento de Control y Evaluación, quien archiva los papeles de trabajo, y con una copia del informe descarga en el inventario las observaciones determinadas, o en su caso, a las que se dio seguimiento.

3.1.1.7 Reportes

La Contraloría Interna, basándose en los formatos preestablecidos, lleva a cabo una serie de reportes trimestrales con la información relativa resultante de auditorías integrales, específicas, de evaluación de programas y de seguimiento.

3.1.1.8 Observaciones

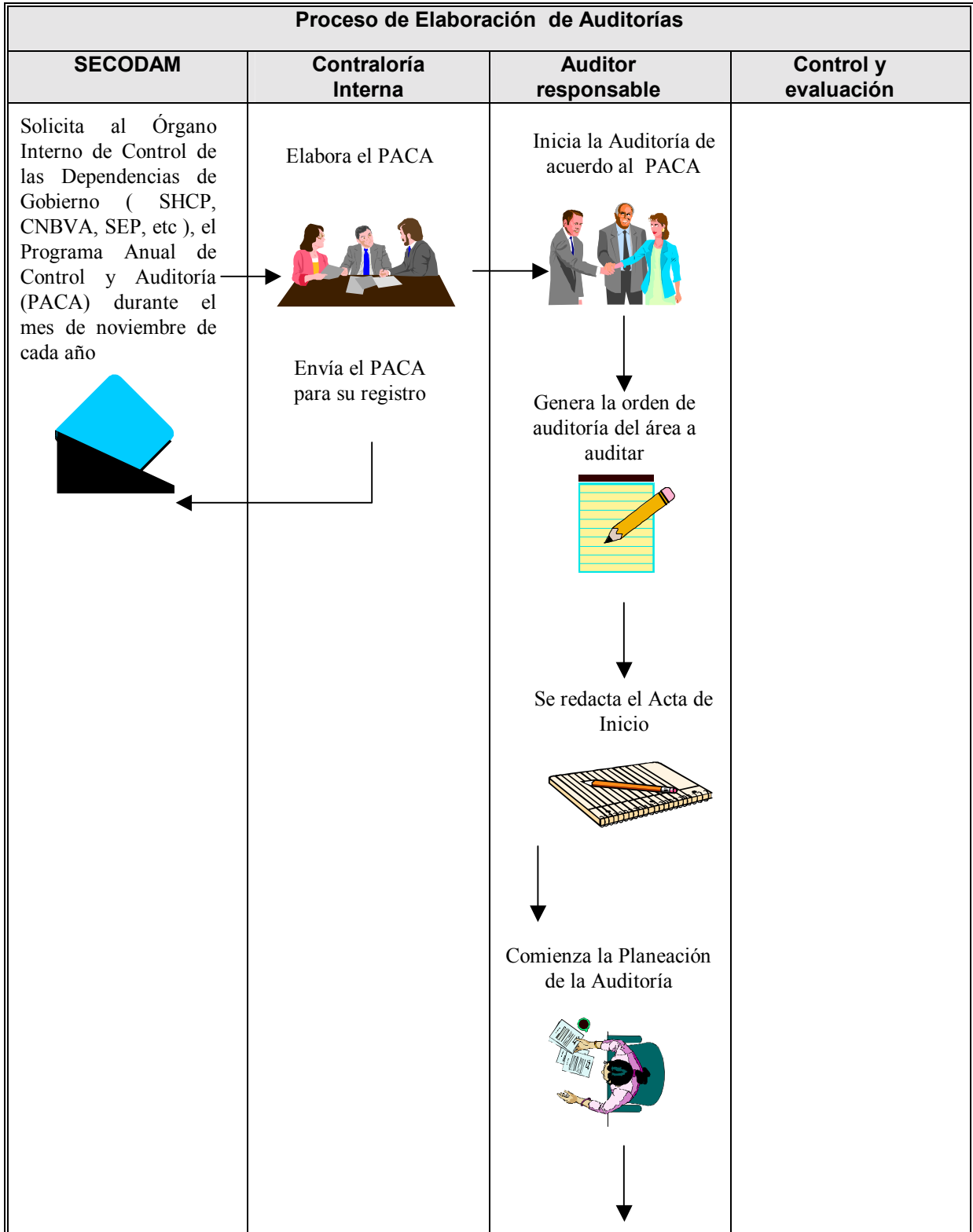
Como resultado de las auditorías practicadas, se determinan las observaciones, que son registradas para su control y seguimiento, haciendo referencia, para su registro, al No. de la Orden de Auditoría y No. Progresivo de Revisión. Cuando la persona encargada de la captura (Control y Evaluación), no conoce bien su referencia –ya sea del No. Progresivo de Revisión o de Orden de auditoría-, entonces puede causar confusión en su identificación, traducándose esto en pérdida de tiempo, y, al no identificarse la referencia utilizada, provocando una serie de errores que afectarían las auditorías pasadas y posteriores.

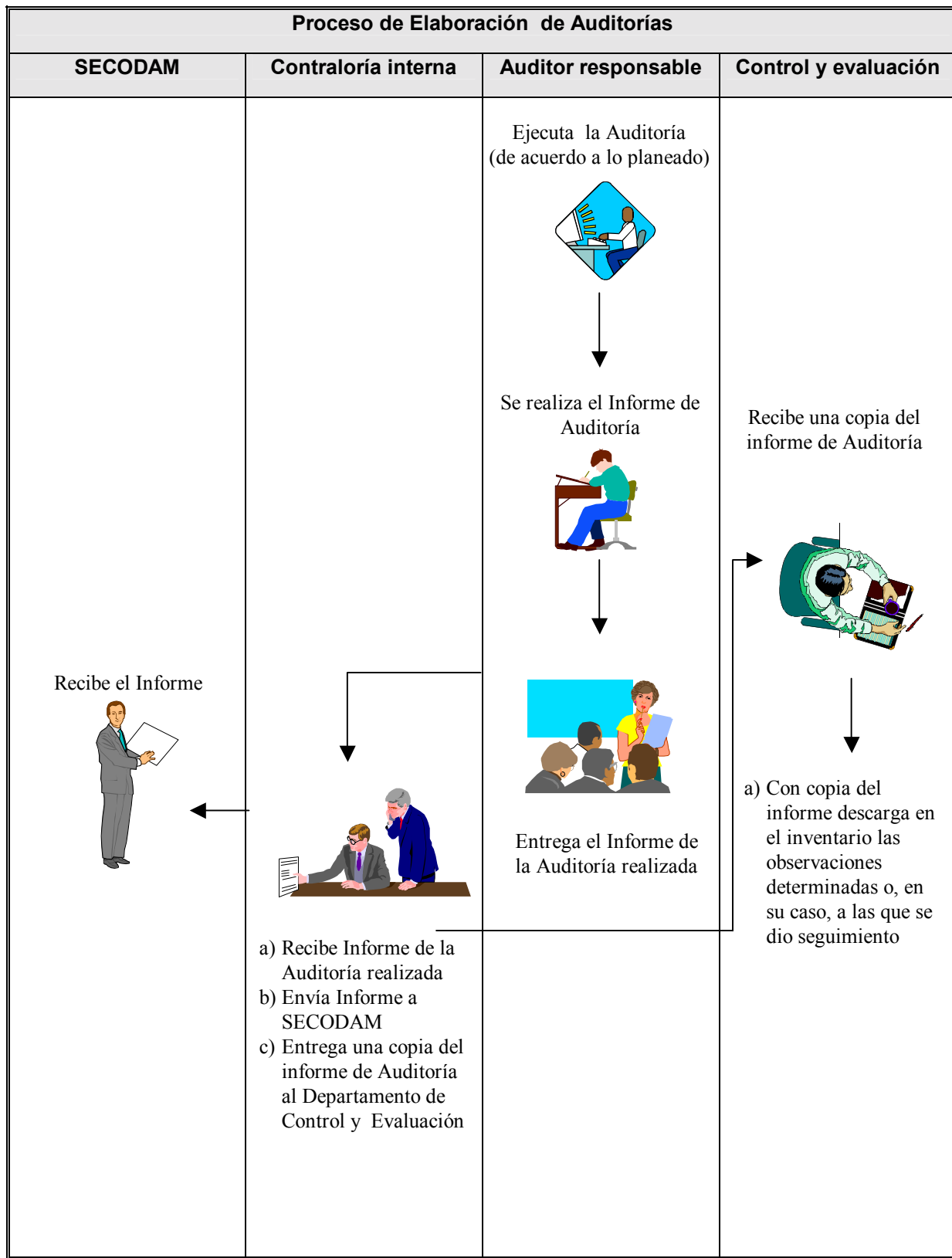
Por otra parte, existe una variedad de reportes (Resumen de las Órdenes de Auditoría, Control de Informes de Auditoría, Control de Informes de Seguimiento, etc.) en que la información, al ser repetitiva, provoca un incremento en la ocupación horas/hombre pues el trabajo se hace en el doble de tiempo.

Cabe mencionar que estos reportes son elaborados utilizando el procesador de texto Word y las hojas de cálculo de Excel, ambos de Microsoft, para la captura, manejo y almacenamiento de la información.

La Contraloría Interna debe obtener los elementos de juicio necesarios para corregir desviaciones dentro de las operaciones de las Direcciones Generales de la institución de gobierno a la que pertenece, y tener los elementos necesarios para promover la corrección de dichas inconsistencias y asegurar una mayor eficiencia en los procedimientos a seguir. Por lo tanto es la Contraloría Interna a quien compete la definición completa de necesidades y evaluación del desempeño del sistema.

Diagrama de Flujo Operativo





3.1.2 Identificación del Problema

Los auditores responsables realizan ciertas actividades en las auditorías de acuerdo con un criterio propio, provocando irregularidades o falta de consistencia en cuanto al manejo de datos; por ejemplo, al hacer referencia de manera distinta (sobre todo en el caso de seguimientos) a las observaciones (unos lo hacen por medio de la Orden de Auditoría y otros por el No. Progresivo de Revisión), se hace difícil detectar casos en los que la Orden de la Auditoría no coincide con el No. Progresivo de Revisión.

Lo anterior provoca confusión y fallas en la continuidad del proceso de descarga en el inventario de observaciones, porque la persona encargada de la captura (Control y Evaluación), no sabe si se está haciendo referencia a la Orden de Auditoría o al Número Progresivo de Revisión, y esto se traduce en pérdida de tiempo e ineficiencia por la dificultad en identificar la falla.

Por otra parte, existe una variedad de reportes (Resumen de las Órdenes de Auditoría, Control de Informes de Auditoría, Control de Informes de Seguimiento, etc.) en que la información es repetitiva, y provoca el incremento en la ocupación horas/hombre.

3.1.3 Objetivos

Los objetivos principales para la creación del Sistema de Automatización del Proceso de Auditoría son los siguientes:

- Generar de manera rápida y eficiente los reportes de auditorías.
- Consulta de datos históricos con precisión y rapidez.
- Rapidez en la captura de la información que estará contenida dentro de los formatos utilizados en las auditorías.
- Reutilización de información entre los diferentes módulos para evitar pérdida de tiempo en la captura.
- Evitar duplicidad en la información.
- Reducción de tiempos en la carga de la información que se capture.
- Minimizar errores en datos.
- Facilitar el seguimiento de las observaciones resultado de las auditorías.
- Automatizar el manejo del inventario de observaciones.

3.2 Análisis de Factibilidad

En el desarrollo de sistemas, el “análisis de factibilidad” es un estudio que sirve para recopilar datos relevantes en los ámbitos operativo, técnico y económico de un proyecto. El fin de dicho estudio es proporcionar información para la alta dirección, y con base en ella, decidir si el desarrollo de un sistema se realiza o no.

La información para el análisis de factibilidad se recopila por medio de entrevistas con los solicitantes del sistema y con los directivos responsables de la toma de decisiones. En el caso de “SAPA” el estudio de factibilidad se llevó a cabo con el Jefe de Departamento de Auditoría en un principio y finalmente con el Director de Supervisión y Auditoría.

3.2.1 Factibilidad Técnica

En la siguiente tabla podemos ver la infraestructura tecnológica mínima necesaria para el desarrollo e implantación de “SAPA”:

Desarrollo de “SAPA”		
Requerimiento	Necesario	Disponible
Sistema operativo	<ul style="list-style-type: none"> Windows 9x 	<ul style="list-style-type: none"> Windows 2000
Software	<ul style="list-style-type: none"> Lenguaje para desarrollo de aplicaciones Windows. Herramienta de diseño de B.D. Editor de Lenguaje SQL 	<ul style="list-style-type: none"> Delphi 5.0 Erwin Isql de SQL Server 7.0
Red	<ul style="list-style-type: none"> Velocidad de transmisión datos 10 Mbps Protocolo Ethernet Topología Común 	<ul style="list-style-type: none"> Velocidad de transmisión datos 100 Mbps Protocolo Ethernet Topología Estrella
DBMS	<ul style="list-style-type: none"> DBMS SQL 	<ul style="list-style-type: none"> SQL Server 7.0
Hardware	<ul style="list-style-type: none"> 2 Computadoras personales • Pentium a 150 Mhz • Memoria RAM: 32 MB • Espacio disco duro: 1 GB • Monitor de 14” 	<ul style="list-style-type: none"> 2 Computadoras personales • Pentium III 900 Mhz • Memoria RAM: 256 MB • Espacio libre disco duro: 3 GB • Monitor de 15”

Tabla 3.1 Factibilidad técnica

Operación de “SAPA” en los clientes		
Requerimiento	Necesario	Disponible
Sistema operativo	<ul style="list-style-type: none"> Windows 9x 	<ul style="list-style-type: none"> Windows 2000
Software	<ul style="list-style-type: none"> Office 97 Drivers SQL DBEngine 	<ul style="list-style-type: none"> Office 2000 Drivers SQL Server 7.0 DBEngine de Delphi 5.0
Red	<ul style="list-style-type: none"> Velocidad de transmisión datos 10 Mbps Protocolo Ethernet Topología Común 	<ul style="list-style-type: none"> Velocidad de transmisión datos 100 Mbps Protocolo Ethernet Topología Estrella
DBMS	<ul style="list-style-type: none"> DBMS SQL 	<ul style="list-style-type: none"> SQL Server 7.0
Hardware	Computadoras personales <ul style="list-style-type: none"> Pentium II a 200 Mhz 64 MB en memoria RAM Espacio disco duro: 500 MB Monitor de 14” Mouse Tarjeta de red Ethernet 	10 Computadoras personales <ul style="list-style-type: none"> Pentium III a 900 Mhz 256 MB en memoria RAM Espacio disco duro: 2 GB Monitor de 14” Mouse Tarjeta de red Ethernet

Tabla 3.2 Operación de “SAPA” en los clientes

Operación de “SAPA” en el Servidor		
Requerimiento	Necesario	Disponible
Sistema operativo	<ul style="list-style-type: none"> Windows NT 	<ul style="list-style-type: none"> Windows 2000 Server
Software		
Red	<ul style="list-style-type: none"> Velocidad de transmisión datos 10 Mbps Protocolo de Red Topología Común 	<ul style="list-style-type: none"> Velocidad de transmisión datos 100 Mbps Protocolo Ethernet Topología Estrella
DBMS	<ul style="list-style-type: none"> DBMS SQL 	<ul style="list-style-type: none"> SQL Server 7.0
Hardware	Servidor <ul style="list-style-type: none"> Pentium II a 300 Mhz 128 MB en memoria RAM Espacio en disco duro de 2 GB 	Compaq Proliant ML 350 <ul style="list-style-type: none"> Procesadores Xeon a 3 GHz 1 GB en memoria RAM 4 discos duros de 40 GB Unidad de cinta dat 8

Tabla 3.3 Operación de “SAPA” en el Servidor

Con base en las tablas anteriores, se pudo concluir que el desarrollo y la implantación de “SAPA” será técnicamente factible con la infraestructura de la se disponía.

3.2.2 Factibilidad operativa

La “factibilidad operativa” es el pronóstico de si el sistema a construirse será realmente empleado por los usuarios que solicitan su desarrollo.

El proceso de auditoría operaba de manera relativamente eficiente porque tenía algunas fallas (comentadas en la sección 3.1.2), debido a estas fallas se perdían horas hombre de trabajo. La optimización del proceso de auditoría por medio de un sistema computacional, fue un importante argumento para su aceptación por parte del personal del área de Contraloría Interna, llamando poderosamente la atención tres puntos: la automatización del “Inventario de observaciones”, la eliminación de confusiones en la referencia a observaciones por Número de Orden de Auditoría o Número Progresivo de Revisión y, por último, la reutilización de los datos capturados en todos los documentos del proceso.

Para que la implantación del sistema tuviese un resultado satisfactorio, fue importante considerar un problema potencial que pudo hacerlo no operable: los usuarios del sistema no estaban familiarizados con el uso de ningún sistema de propósito específico, sólo empleaban las aplicaciones de uso más común en oficina (Word) y su conocimiento del sistema operativo (Windows) es, en general, de nivel básico por lo que fue necesario considerar una buena capacitación del personal para que el sistema no fuera subutilizado.

Debido al gran interés presentado por el área y gracias a la correcta detección de las necesidades fundamentales de los usuarios con respecto a la optimización del proceso, se pudo determinar que la futura operación de “SAPA” sería factible.

3.2.3 Factibilidad Económica

En el estudio de la factibilidad económica, se toman en cuenta tres factores principalmente: costo del desarrollo (tiempo, tipo de personal, hardware y software), costo del tiempo empleado por la empresa y costo del equipo necesario para su implantación y operación.

Para el desarrollo del sistema sólo fue necesario proponer la compra de una herramienta de programación (*Delphi*); el resto de los requerimientos, -Windows NT,95 y/o 98, SQL Server, Erwin y Office-, fueron proporcionados tanto por el área de Contraloría Interna como por el área de Sistemas de la institución de Gobierno. Los requerimientos de hardware para desarrollo e implantación también fueron proporcionados por ambas áreas por lo que no fue necesario comprar ningún equipo.

La parte más costosa del proyecto fue la propuesta de contratación de los recursos humanos necesarios: un líder de proyecto y un programador/analista. A pesar de ser la parte más costosa, la inversión en recursos humanos para el desarrollo e implementación de “SAPA” se estimó recuperable en un período máximo de dos años.

El costo del tiempo empleado por el área de contraloría para el desarrollo e implantación es reducido debido que la planeación de las actividades de análisis (entrevistas), revisión de avances, capacitación y pruebas, está organizada de manera que no afecten significativamente el trabajo desempeñado en el área por los usuarios que participen en las diferentes actividades del proyecto; de hecho, los beneficios que proporciona “SAPA” en la optimización de tiempos del proceso hace que la inversión de costo/tiempo inicial sea recuperada aproximadamente en el primer año de operación de “SAPA”.

De lo anterior se dedujo que el sistema “SAPA” será económicamente factible.

3.3 Análisis de Requerimientos de la Información

3.3.1 Entrevistas

MINUTA DE ENTREVISTA Proyecto: “SAPA” (Sistema de Automatización del Proceso de Auditoría) Área: Contraloría Interna	Fecha: 01/06/2003	Minuta No. 1
	Hora de inicio: 9:00 a.m.	Terminó: 11:00 a.m.

No.	AGENDA	Duración
Asunto		
1	Definición general del sistema actual	1 hr.
2	Detección de problemática	1 hr

PERSONAS CONVOCADAS	
PRESENTES	ABREVIATURA
Lic. Antonio Jiménez Pérez (Director de Supervisión y Auditoría)	AJ
C.P. Luis González Torres (Jefe de Departamento de Auditoría)	LG
Ing. Melvin Romero Gómez (Subdirector de Operación de Sistemas)	MR
C. Antonio Cejudo Ibarra (Coordinador en Reingeniería)	AC

No. Asunto	HECHOS
1	<p>(LG) Realizó una explicación general a (MR) y (AC) de cómo se desarrolla el proceso de auditoría:</p> <p>La Secretaría de Contraloría y Desarrollo Administrativo (SECODAM) solicita al Órgano Interno de Control el Programa Anual de Control de Auditorías (PACA) durante el mes de noviembre para tener conocimiento de las auditorías a realizar el próximo año dentro de la institución de gobierno. La Dirección de Supervisión y Auditoría elabora el PACA, lo envía a la SECODAM para su registro y da una copia al auditor responsable que inicia la auditoría, mismo que genera la orden de auditoría del área a auditar, también redacta el acta de inicio y posteriormente comienza la planeación de la auditoría, donde se realiza la carta planeación y se elabora el cronograma de actividades a desarrollar durante la misma, inmediatamente se ejecuta la auditoría de acuerdo a lo planeado, incluyéndose: el marco conceptual, que se elabora mientras se desahoga y donde se revisa por medio de cruces la coherencia de la información; y las cédulas de auditoría o seguimiento, según sea el caso, para posteriormente realizar el Informe de Auditoría donde se incluirá la cédula única de auditoría y la evaluación de esta.</p> <p>Una vez realizado el informe, se entrega a la dirección de Contraloría junto con los papeles de trabajo, la cual envía el informe a SECODAM y una copia del informe y los papeles de trabajo al Departamento de Control y Evaluación, quien a su vez archiva los papeles de trabajo y, con una copia del informe, descarga en el inventario las observaciones determinadas, o en su caso, a las que se dio seguimiento; además se captura en el Sistema de Información Periódica (SIP) las observaciones relevantes o las que resultaron atendidas.</p> <p>La dirección de Contraloría Interna, con base en los formatos preestablecidos, lleva a cabo una serie de reportes con información relativa resultante de auditorías integrales, específicas, de evaluación de programas y de seguimiento. La información contenida en estos reportes resultante de la auditoría es procesada por el SIP, de donde se obtienen los informes que serán mandados a SECODAM vía Internet.</p>
2	<p>(LG) y (AJ) exponen a (MR) y (AC) la problemática actual de la siguiente forma:</p> <p>Los auditores responsables realizan ciertas actividades en las auditorías de acuerdo con un criterio propio, mismas que han provocado irregularidades o falta de consistencia en cuanto al manejo de datos, al hacer referencia de manera distinta -sobretudo en el caso de seguimientos-, a las observaciones, ya que unos lo hacen por medio de la Orden de Auditoría y otros por el No. Progresivo de Revisión, que, al no coincidir, hacen difícil su detección, provocando confusión y fallas en la continuidad del proceso de descarga en el inventario de observaciones. Como consecuencia se da una pérdida de tiempo y eficiencia.</p> <p>Por otra parte, existe una variedad de reportes (Resumen de las Órdenes de Auditoría, Control de Informes de Auditoría, Control de Informes de Seguimiento, etc.) en donde la información es repetitiva, provocando el incremento en la ocupación horas/hombre.</p> <p>Cabe mencionar que estos reportes son elaborados utilizando Word y Excel capturando la información manualmente.</p>

ACCIONES A SEGUIR			
No. Asunto	DESCRIPCIÓN	Responsable	Fecha de compromiso
2	Con objeto de conocer de manera más concreta la problemática del sistema, (MR) y (AC) solicitan a (AJ) y (LG) los documentos que serán objeto de automatización así como una descripción de la función y uso de los mismos.	(AJ) y (LG)	08/06/2003
2	(MR) y (AC) se comprometen a realizar un análisis inicial de la información para determinar los objetivos que debe cumplir el sistema para poder enfrentar los problemas detectados en el sistema actual.	(MR) y (AC)	08/06/2003

MINUTA DE ENTREVISTA Proyecto: "SAPA" (Sistema de Automatización del Proceso de Auditoría) Área: Contraloría Interna	Fecha: 08/06/2003	Minuta No. 2
	Hora de inicio: 9:00 a.m.	Terminó: 11:00 a.m.

No.	AGENDA	Duración
3	Definición de objetivos de "SAPA"	30 min.
4	Documentos e informes a generar	30 min.
5	Revisión de metodología de llenado de los documentos	1 hr.

PERSONAS CONVOCADAS	
PRESENTES	ABREVIATURA
Lic. Antonio Jiménez Pérez (Director de Supervisión y Auditoría)	AJ
C.P. Luis González Torres (Jefe de Departamento de Auditoría)	LG
Ing. Melvin Romero Gómez (Subdirector de Operación de Sistemas)	MR
C. Antonio Cejudo Ibarra (Coordinador en Reingeniería)	AC

No.	HECHOS
3	<p>(MR) y (AC) presentan los objetivos desarrollados a partir de la información proporcionada en la entrevista anterior:</p> <p>Se da la necesidad de desarrollar un sistema computacional que permita la automatización en la creación de reportes, así como de un nuevo inventario de observaciones y su seguimiento utilizando la información que se encuentra en documentos en Excel y Word.</p> <p>Para agilizar el proceso de una auditoría, el usuario podrá crear los documentos necesarios para desarrollar el proceso de manera rápida y sencilla, ya que se tendrá una opción para esto. Se contará con un formato para cada uno de los documentos donde se tendrá que llenar los espacios que cambian en cada auditoría.</p> <p>En atención a la necesidad de confiabilidad, continuidad y control en la captura de información, es pertinente evitar repetir datos durante este proceso, para ello los bloques de información repetidos en cada documento serán sólo introducidos una vez, de este modo los títulos y encabezados de las cédulas que utilicen los mismos datos aparecerán automáticamente.</p> <p>Los objetivos pueden entonces definirse como sigue:</p> <ul style="list-style-type: none"> ▪ Generar de manera rápida y eficiente los informes de auditorías. ▪ Consulta de datos históricos con precisión y rapidez. ▪ Rapidez en la captura de la información contenida dentro de los formatos utilizados en las auditorías. ▪ Reutilización de información entre los diferentes módulos para evitar tiempo perdido en la captura. ▪ Evitar duplicidad en la información. ▪ Reducción de tiempos en la carga de la información que se capture. ▪ Minimizar errores en datos.

	<ul style="list-style-type: none"> ▪ Facilitar el seguimiento de las observaciones resultado de las auditorías. <p>(AJ) y (LG) exponen que los objetivos coinciden con las necesidades expresadas en la entrevista anterior pero hacen énfasis en la necesidad de automatizar el manejo del inventario de observaciones.</p>
4	<p>(AJ) y (LG) hacen entrega de la guía de auditoría a (MR) y (AC). Esta guía contiene todos los documentos que se emplean en el desarrollo de la auditoría, así como la manera y el momento en la que son llenados.</p> <p>(AJ) y (LG) dan una breve explicación de los documentos:</p> <p><i>PACA Resumen.</i>- Es donde se registran todas las auditorías a realizarse en el año corriente. Su información esta resumida.</p> <p><i>PACA Detallado.</i>- es donde se registran todas las auditorías a realizarse en el año corriente. Se presenta el detalle de la información.</p> <p><i>Orden de Auditoría.</i>- El Órgano interno de control (contraloría interna) informa al responsable del área a auditar la próxima realización de una auditoría.</p> <p><i>Acta de inicio.</i>- Es donde se especifican las condiciones específicas del inicio de una auditoría.</p> <p><i>Carta planeación.</i>- Se define la forma en la que se llevará a cabo la auditoría (antecedentes, objetivo, estrategia, etc.).</p> <p><i>Cronograma de actividades.</i>- Es la calendarización de las actividades de la auditoría.</p> <p><i>Marco conceptual.</i>- Contiene información sobre la manera de resolver las observaciones generadas en la auditoría.</p> <p><i>Cédulas de observaciones</i> (nuevas y de seguimiento).- Aquí se asientan las observaciones detectadas en el área auditada.</p> <p><i>Informe de auditoría.</i>- Se recopila la información de la auditoría y se construye un informe que se entrega al responsable del área auditada.</p> <p><i>Cédula única de auditoría.</i>- Contiene un resumen de información de la auditoría, sus observaciones y los montos de las mismas.</p> <p><i>Evaluación de auditoría.</i>- Formato para evaluar la auditoría.</p>
5	<p>(AJ) y (LG) explican a (MR) y (AC) la forma de usar la guía de auditoría. El formato es el siguiente:</p> <ul style="list-style-type: none"> ▪ Primero la carátula que indica el documento a emplear. ▪ Guía de llenado de los documentos (con una referencia numérica de datos específicos). ▪ Formato específico en blanco con referencia numérica a la guía de llenado.

ACCIONES A SEGUIR			
No. Asunto	DESCRIPCIÓN	Responsable	Fecha de compromiso
4	(MR) y (AC) solicitan a (AJ) y (LG) ejemplos reales de los documentos que se deben llenar con la guía de auditoría.	(AJ) y (LG)	15/06/2003
5	(MR) y (AC) analizarán la guía de auditoría para poder definir con mayor precisión los módulos del sistema.	(MR) y (AC)	15/06/2003

MINUTA DE ENTREVISTA Proyecto: "SAPA" (Sistema de Automatización del Proceso de Auditoría) Área: Contraloría Interna	Fecha: 15/06/2003	Minuta No. 3
	Hora de inicio: 9:00 a.m.	Terminó: 11:00 a.m.

No.	AGENDA	Duración
Asunto		
6	Definición de los módulos del sistema	30 min.
7	Definición de PACA	30 min.
8	Definición de Orden de auditoría	30 min.
9	Definición de inventario de observaciones	30 min.

PERSONAS CONVOCADAS	
PRESENTES	ABREVIATURA
C.P. Luis González Torres (Jefe de Departamento de Auditoría)	LG
C. Antonio Cejudo Ibarra (Coordinador en Reingeniería)	AC

No.	HECHOS
Asunto	
6	<p>(AC) presenta a (LG) la definición de los módulos del sistema basados en la necesidad de generar los documentos de proceso de auditoría:</p> <ul style="list-style-type: none"> ▪ PACA ▪ Orden de Auditoría ▪ Acta de inicio ▪ Carta planeación ▪ Cronograma de actividades ▪ Marco conceptual ▪ Cédulas de observaciones ▪ Cédulas de seguimiento ▪ Informe de auditoría ▪ Cédula única de auditoría ▪ Reportes ▪ Consulta de inventario de observaciones ▪ Catálogos <p>Los módulos propuestos se definieron con base en el proceso de elaboración de auditorías para facilidad en el manejo del sistema por parte del usuario. (LG) esta de acuerdo con los módulos planteados a desarrollar y comenta la necesidad de colaborar en la definición de las necesidades de desarrollo de estos módulos.</p>
7	<p>(LG) explica a detalle el funcionamiento del módulo PACA basándose en la guía de auditoría y en los ejemplos del PACA que entrega en esta junta a (AC): El PACA (Programa Anual de Control y Auditoría) es un documento en el cual se asientan las diferentes auditorías a realizarse durante un año. La referencia de las auditorías es a través de un número progresivo de revisión, lo que más adelante puede provocar confusión. Los datos año, fecha, no. prog. de revisión, clave del programa y título, pertenecen a una sola revisión. Una revisión es un tipo de auditoría a llevar a cabo. Estos datos son los que se muestran en</p>

	<p>el reporte de PACA Resumen. Cada revisión tiene diferentes rubros. Un rubro es la definición de diferentes ámbitos en las cuales se puede llevar a cabo una auditoría. Cada rubro tiene diferentes descripciones de la revisión. Las descripciones de la revisión definen a detalle lo que se pretende alcanzar al llevar a cabo una revisión a alguna área.</p> <p>Esta organización de la información aparece en el ejemplo de PACA Resumen y PACA Detallado y es fácil de identificar.</p> <p>Para auxiliarnos en los requerimientos de datos, su definición y la forma en que se capturan en el documento, la guía de auditoría y los ejemplos de los documentos (anexos a la minuta), son la base.</p> <p>(LG) y (AC) definen los datos que son redundantes y los que son nuevos en la captura del documento:</p> <ul style="list-style-type: none"> ▪ Año del PACA (mismo de PACA resumen) ▪ Fecha de Elaboración del PACA (mismo de PACA resumen) ▪ No Progresivo de Revisión (mismo de PACA resumen) ▪ Clave del programa (mismo de PACA resumen) ▪ Título de la Revisión (mismo del PACA Resumen) ▪ Rubro (Nuevo dato) ▪ Objetivos (Nuevo dato) ▪ Descripción de la Revisión Detallada (Nuevo dato) ▪ Firma del titular OIC (Fijo)
8	<p>(LG) explica a detalle el funcionamiento del módulo Orden de Auditoría basándose en la guía de auditoría y en los ejemplos de la misma que entrega en esta junta a (AC) :</p> <p>La orden de auditoría es el documento donde se aclara, con base en fundamentos legales, la revisión que se llevará a cabo en el área especificada; también se menciona de manera general los alcances de la auditoría y el período por revisar, así como el tiempo en el que se practicará la revisión. Para el caso de las auditorías de seguimiento se mencionarán los antecedentes de las revisiones donde surgieron las observaciones sujetas a seguimiento.</p> <p>Para auxiliarnos en los requerimientos de datos, su definición y la forma en que se capturan en el documento, la guía de auditoría y los ejemplos de los documentos (anexos a la minuta), son la base.</p> <p>(LG) y (AC) definen qué datos son redundantes, nuevos y de catálogo en la captura del documento resultando:</p> <ul style="list-style-type: none"> ▪ No. de orden de auditoría (Nuevo dato) ▪ Fecha de elaboración de la orden de auditoría (Nuevo dato) ▪ Fecha de inicio del período por revisar (Nuevo dato) ▪ Fecha final del período por revisar (Nuevo dato) ▪ Fecha de inicio de la duración de la revisión (Nuevo dato) ▪ Fecha de final de la duración de la revisión (Nuevo dato) ▪ Objeto de la revisión (Nuevo dato) ▪ No. progresivo de revisión (Viene del PACA) ▪ Clase (Tipo) de auditoría (Se elige de catálogo) ▪ Área por auditar (Se elige de catálogo) ▪ Nombre de Auditores responsables de la revisión (son 2, uno se elige de catálogo, y el otro es el director de área y es fijo) ▪ Nombre, cargo y piso del director general o vicepresidente (según sea el caso) del área auditada (Se elige de catálogo) ▪ Nombre y cargo de personal a quién se dirigen copias. (Se elige de catálogo) ▪ Nombre del titular del OIC (Órgano Interno de Control) (Fijo)

9	<p>(LG) explica a (AC) la forma en la que se maneja el inventario de observaciones:</p> <p>Las observaciones surgen a partir de la ejecución de una auditoría y definen la problemática que posee el área auditada. Una auditoría puede generar varias observaciones. Las observaciones también pueden definirse como los problemas o faltas específicas en las que incurre un área y que hacen que su operación no sea la adecuada.</p> <p>Las observaciones generadas durante una orden de auditoría pueden ser atendidas (resueltas) durante el desarrollo de la misma, o quedar pendientes. Al quedar pendientes las observaciones, se tendrán que generar nuevas auditorías que les den seguimiento.</p> <p>Las observaciones pendientes tienen un grado de avance y solventación (solución) de la problemática que las generó. Si el avance en la solventación es menor al 100 %, estas observaciones requerirán de nuevas auditorías que las atiendan hasta que su grado de solventación sea del 100% (es decir, que sean completamente atendidas).</p> <p>Al conjunto de observaciones que no han sido solventadas se les conoce como el inventario de observaciones, y de él parten los auditores para generar las auditorías de seguimiento.</p> <p>Una observación es definida por el (los) auditor(es) responsable(s) de la auditoría durante la misma. Después de redactarla, el auditor la somete a revisión por parte de la Dirección de Supervisión y Auditoría y luego a discusión con el área auditada. Después de este procedimiento las observaciones son eliminadas (atendidas) y/o corregidas de ser necesario, y son firmadas como observación definitiva por el Director de Supervisión y Auditoría. Es entonces cuando una observación pasa al formar parte del inventario.</p> <p>Para que esta observación salga del inventario será necesario que se le dé seguimiento en otra auditoría y se definan un avance de solventación del 100% y un estatus de solventada, esto puede tomar varias auditorías de seguimiento (no deberían de ser más de tres).</p>
---	--

ACCIONES A SEGUIR			
No. Asunto	DESCRIPCIÓN	Responsable	Fecha de compromiso
10 11 12 13	(AC) se compromete a analizar las siguientes definiciones desde la guía de auditoría: Definición de Acta de inicio Definición de Carta planeación Definición de Cronograma de actividades Definición de Marco conceptual	(AC)	22/06/2003
9	(AC) se compromete a realizar un diagrama de flujo para representar el manejo del inventario.	(AC)	22/06/2003

MINUTA DE ENTREVISTA Proyecto: “SAPA” (Sistema de Automatización del Proceso de Auditoría) Área: Contraloría Interna	Fecha: 22/06/2003	Minuta No. 4
	Hora de inicio: 9:00 a.m.	Terminó: 11:00 a.m.

No.	AGENDA	Duración
Asunto		
10	Definición de Acta de inicio	30 min.
11	Definición de Carta planeación	30 min.
12	Definición de Cronograma de actividades	30 min.
13	Definición de Marco conceptual	30 min.

PERSONAS CONVOCADAS	
PRESENTES	ABREVIATURA
C.P. Luis González Torres (Jefe de Departamento de Auditoría)	LG
C. Antonio Cejudo Ibarra (Coordinador en Reingeniería)	AC

No.	HECHOS
Asunto	
10	<p>(LG) explica a detalle el funcionamiento del módulo Acta de Inicio a (AC): El Acta de Inicio es el documento que narra las circunstancias específicas del inicio de una auditoría, por lo que en su elaboración deben guardarse las formalidades correspondientes. Para auxiliarnos en los requerimientos de datos, su definición y la forma en que se capturan en el documento, la guía de auditoría y los ejemplos de los documentos (anexos a la minuta), son la base.</p> <p>(LG) y (AC) definen los datos redundantes y los nuevos en la captura del documento resultando:</p> <ul style="list-style-type: none"> ▪ Números de Acta (0 + No. Orden + Últimos 2 dígitos del año de la orden + No. de hoja del acta con formato 00#) ▪ No de Orden de Auditoría (Viene de Orden de Auditoría) ▪ Área y su ubicación por auditar (Viene de Orden de Auditoría) ▪ Fecha de elaboración de orden de auditoría (Viene de Orden de Auditoría) ▪ Período a revisar (viene de Orden de auditoría) ▪ Objeto de la revisión (viene de Orden de auditoría) ▪ Duración de la revisión (viene de Orden de Auditoría) ▪ Nombre y credencial de auditores responsables (Viene de orden de Auditoría) ▪ Nombre, credencial, edad, estado civil, origen, domicilio y RFC del servidor público responsable por auditar (Es el director del área y se elige de catálogo) ▪ Nombre, domicilio, credencial, RFC de los testigos (Se elige de catálogo) ▪ Fecha de Elaboración de acta de inicio (Nuevo Dato) ▪ Hora inicial de hechos (Nuevo dato) ▪ Hora final de hechos (Nuevo dato)
11	<p>(LG) explica a detalle el funcionamiento del módulo Carta planeación a (AC): La Carta planeación es el documento donde se hace mención de los antecedentes del área programada o rubro por auditar, describiendo su estructura, marco legal aplicable, principales</p>

	<p>políticas y funciones, actividades que realiza, distribución física de las áreas de trabajo, resultados más importantes en auditorías anteriores, información financiera, presupuestaria y aquellos elementos que permitan al auditor tener un conocimiento general antes de iniciar la auditoría, así como la finalidad y alcances de la revisión. Para auxiliarnos en los requerimientos de datos, su definición y la forma en que se capturan en el documento, la guía de auditoría y los ejemplos de los documentos (anexos a la minuta), son la base.</p> <p>(LG) y (AC) definen los datos redundantes y los nuevos en la captura del documento resultando:</p> <ul style="list-style-type: none"> ▪ No de orden de Auditoría (Viene de Orden de Auditoría) ▪ Tipo (clase) de auditoría (Viene de Orden de Auditoría) ▪ Área por Auditar (Viene de Orden de Auditoría) ▪ Objetivos Establecidos (Viene de PACA) ▪ Período a Revisar (Viene del Acta de Inicio Fecha 1 y Fecha 2) ▪ Nombre e iniciales del personal comisionado (Viene de Orden de Auditoría y son Auditor Responsable 1 y Auditor Responsable 2) ▪ Fecha de Elaboración de Carta Planeación (Dato Nuevo) ▪ Trimestre (Dato Nuevo) ▪ Antecedentes del área (Dato Nuevo) ▪ Alcance de los aspectos de Revisión (Dato Nuevo) ▪ Problemática (Dato Nuevo) ▪ Estrategia (Dato Nuevo)
12	<p>(LG) explica a detalle el funcionamiento del módulo Cronograma de actividades a (AC): El Cronograma de actividades es el documento donde se refiere a las acciones que de manera general se ejecutarán durante el tiempo programado de la revisión, como son: inicio de la auditoría, levantamiento del acta, entrevistas con servidores públicos, solicitud de información, visitas físicas a instalaciones u oficinas, período de análisis del área, programa o rubro, comentario de observaciones y elaboración del informe entre otros.</p> <p>Para auxiliarnos en los requerimientos de datos, su definición y la forma en que se capturan en el documento, la guía de auditoría y los ejemplos de los documentos (anexos a la minuta), son la base.</p> <p>(LG) y (AC) definen los datos redundantes y los nuevos en la captura del documento resultando:</p> <ul style="list-style-type: none"> ▪ Área Auditada (Viene de Orden de Auditoría) ▪ Número Progresivo de Revisión (Viene de Orden de Auditoría) ▪ Tipo (Clase) de Auditoría (Viene de Orden de Auditoría) ▪ Nombre e iniciales del personal comisionado (Viene de la orden de Auditoría) <p>No van a la base de datos</p> <ul style="list-style-type: none"> ▪ Actividad (Nuevo Dato) ▪ Mes de Actividades (Nuevo Dato) ▪ Número consecutivo de Actividad (Nuevo Dato) ▪ Descripción de la Actividad (Nuevo Dato) ▪ Estimado de Días Totales de Revisión (Nuevo Dato) <p>Los datos que no van a la base de datos, son definidos de esta forma porque es información cuyo almacenamiento no es relevante.</p>

13	<p>(LG) explica a detalle el funcionamiento del módulo Marco Conceptual a (AC): El Marco conceptual es el documento que describe las cifras totales de las operaciones que se analizarán, tanto las financieras como las físicas. En todos los casos debe mencionarse la situación de los datos o cifras que se están manejando, los criterios utilizados para seleccionar la muestra, y, además, las técnicas específicas a desarrollar y los procedimientos que llevará a cabo el auditor para cumplir con su objetivo. El desglose de los procedimientos debe tener un orden cronológico y ser congruente con las operaciones a revisar, se desagregarán tantos procedimientos como profundidad se dé al aspecto por revisar. Para auxiliarnos en los requerimientos de datos, su definición y la forma en que se capturan en el documento, la guía de auditoría y los ejemplos de los documentos (anexos a la minuta), son la base. (LG) y (AC) definen los datos redundantes y los nuevos en la captura del documento resultando:</p> <ul style="list-style-type: none"> ▪ No de orden de auditoría (Viene del Orden de Auditoría) ▪ No progresivo de revisión (Viene del Orden de Auditoría) ▪ Área a auditar (Viene de Orden de Auditoría) ▪ Clase de auditoría (Viene del Orden de Auditoría) ▪ Iniciales de auditor (Viene del Orden de Auditoría) ▪ No. de Cédula (Nuevo Dato) ▪ Fecha de Marco Conceptual (Nuevo Dato) ▪ Objetivo (Nuevo Dato) ▪ Universo (Nuevo Dato) ▪ Alcance (Nuevo Dato) ▪ Fuente (Nuevo Dato) ▪ Procedimiento (Nuevo Dato) ▪ Conclusión (Nuevo Dato)
----	---

ACCIONES A SEGUIR			
No. Asunto	DESCRIPCIÓN	Responsable	Fecha de compromiso
	(AC) se compromete a analizar las siguientes definiciones desde la guía de auditoría:	(AC)	29/06/2003
15	Definición de Cédula de observaciones		
16	Definición de Cédula de seguimientos		
17	Definición de Informe de auditoría		
18	Definición de Cédula única		
19	Definición de Evaluación de auditoría		
20	Definición de Reportes		

Sistema de Automatización del Proceso de Auditoría

MINUTA DE ENTREVISTA Proyecto: "SAPA" (Sistema de Automatización del Proceso de Auditoría) Área: Contraloría Interna	Fecha: 29/06/2003	Minuta No. 5
	Hora de inicio: 9:00 a.m.	Terminó: 12:00 p.m.

No.	AGENDA	Duración
	Asunto	
15	Definición de Cédula de observaciones	30 min.
16	Definición de Cédula de seguimientos	30 min.
17	Definición de Informe de auditoría	30 min.
18	Definición de Cédula única	30 min.
19	Definición de Evaluación de auditoría	30 min.
20	Definición de Reportes	30 min.

PERSONAS CONVOCADAS	
PRESENTES	ABREVIATURA
C.P. Luis González Torres (Jefe de Departamento de Auditoría)	LG
C. Antonio Cejudo Ibarra (Coordinador en Reingeniería)	AC

No.	HECHOS
15	<p>(LG) explica a detalle el funcionamiento del módulo Cédula de observaciones a (AC): La Cédula de observaciones es el documento en el que se depositan todos los datos de las observaciones generadas durante la auditoría. Para auxiliarnos en los requerimientos de datos, su definición y la forma en que se capturan en el documento, la guía de auditoría y los ejemplos de los documentos (anexos a la minuta), son la base. (LG) y (AC) definen los datos redundantes y los nuevos en la captura del documento resultando:</p> <ul style="list-style-type: none"> ▪ No de orden de auditoría (Viene de Orden de Auditoría) ▪ Área auditada (Viene de Orden de Auditoría) ▪ Clase de auditoría (Viene de Orden de Auditoría) ▪ Número de observación (Nuevo Dato) ▪ Clasificación de la observación (Nuevo Dato) ▪ Título de la observación (Dato Nuevo) ▪ Irregularidad (Nuevo Dato) (Observación) ▪ Causa (Nuevo Dato) ▪ Efecto (Nuevo Dato) ▪ Fundamento legal (Dato Nuevo) ▪ Recomendación correctiva (Nuevo Dato) ▪ Recomendación preventiva (Nuevo Dato) ▪ Fecha de discusión de la observación (Nuevo Dato) ▪ Fecha de implementación (Nuevo Dato) ▪ Monto fiscalizable (Nuevo Dato) ▪ Monto fiscalizador (Nuevo Dato) ▪ Monto Irregularidad (Nuevo Dato)

	<ul style="list-style-type: none"> ▪ Nombre y cargo del servidor público Responsable por auditar (Es el director del área y viene de la orden de auditoría)
16	<p>(LG) explica a detalle el funcionamiento del módulo Cédula de seguimientos a (AC): La Cédula de seguimientos es el documento en el cuál se depositan todos los datos de las observaciones que han pasado por algún seguimiento, así como sus replanteamientos. Para auxiliarnos en los requerimientos de datos, su definición y la forma en que se capturan en el documento, la guía de auditoría y los ejemplos de los documentos (anexos a la minuta), son la base. (LG) y (AC) definen los datos redundantes y los nuevos en la captura del documento resultando:</p> <ul style="list-style-type: none"> ▪ No de orden de Auditoría (Viene de Orden de Auditoría) ▪ Número de observación (Viene de cédula de observaciones) ▪ Área auditada (Viene de Orden de Auditoría) ▪ Clase de auditoría (Viene de Orden de Auditoría) ▪ Observación original (Viene de Cédula observaciones de los campos título de observación, irregularidad, recomendación correctiva y recomendación preventiva) ▪ Situación actual (Dato Nuevo) ▪ Recomendaciones preventivas (Replanteamiento) (Dato Nuevo) ▪ Recomendaciones correctivas (Replanteamiento) (Dato Nuevo) ▪ Fecha de discusión de la observación (Nuevo Dato) ▪ Fecha de implementación de la observación (Nuevo Dato) ▪ Solventada (Nuevo Dato) ▪ No solventada (Nuevo Dato) ▪ Avance (Porcentaje) (Nuevo Dato) ▪ Monto fiscalizable(Nuevo Dato) ▪ Monto fiscalizador(Nuevo Dato) ▪ Monto irregularidad(Nuevo Dato) ▪ Nombre y cargo del servidor público responsable por auditar (Es el director del área y viene de la orden de auditoría) <p>Cuando se realiza un seguimiento a una observación su recomendación preventiva y correctiva (sean nuevas o de un replanteamiento) pasan a ser parte del campo observación original. De esta manera se genera un nuevo replanteamiento con una nueva recomendación preventiva y correctiva.</p>
17	<p>(LG) explica a detalle el funcionamiento del módulo Informe de auditoría a (AC): El Informe de auditoría es el documento donde se depositan todos los datos de la auditoría realizada. Una vez realizado el informe, y revisado por el Director de Supervisión y Auditoría se entrega al Contralor junto con los papeles de trabajo, quien envía el informe a SECODAM con copia al Presidente de la institución de gobierno, al titular de Unidad Administrativa auditada y otra junto con los papeles de trabajo al Departamento de Control y Evaluación, que a su vez archiva los papeles de trabajo, y con una copia del informe descarga en el inventario las observaciones determinadas, o en su caso, a las que se dio seguimiento. Para auxiliarnos en los requerimientos de datos, su definición y la forma en que se capturan en el documento, la guía de auditoría y los ejemplos de los documentos (anexos a la minuta), son la base.</p>

	<p>(LG) y (AC) definen los datos redundantes y los nuevos en la captura del documento resultando:</p> <ul style="list-style-type: none"> ▪ No de Orden de auditoría (Viene de Orden de Auditoría) ▪ No. Progresivo de revisión (Viene de Orden de Auditoría) ▪ Área auditada (Viene de Orden de Auditoría) ▪ Clase de auditoría (Viene de Orden de Auditoría) ▪ Período de revisión (Viene de Orden de Auditoría) ▪ Duración (Fecha de Inicio y Fecha de Conclusión) (Viene de Orden de Auditoría) ▪ Nombre de Auditores Responsables (Viene de Orden de Auditoría y es sólo uno) ▪ Nombre, cargo y ubicación del Personal Auditado (Viene de orden de auditoría y es la atención de orden de auditoría) ▪ Antecedentes (Viene de la Carta Planeación) ▪ Fecha de discusión de la observación (Viene de la Cédula de observaciones) ▪ Total de observaciones determinadas en la orden (Viene de la Cédula de observaciones) ▪ Objetivo (Viene del PACA) ▪ Número de Oficio (Nuevo Dato) ▪ Fecha del informe (Nuevo Dato) ▪ Informe Ejecutivo (Nuevo Dato) ▪ Alcance (Nuevo Dato) ▪ Resultado del trabajo (Nuevo Dato) ▪ Conclusión y Recomendación General (Nuevo Dato) ▪ Nombre y cargo de personal que recibe copias (Se elige de catálogo)
<p>18</p>	<p>(LG) explica a detalle el funcionamiento del módulo Cédula de única a (AC) :</p> <p>La Cédula de única es el documento donde se lleva un control de los días/hombre utilizados en la realización de la auditoría, así como la cantidad de observaciones determinadas en la revisión; para las Auditorías Integrales deberá anotarse la cantidad de observaciones determinadas por cada rubro; en caso de una auditoría de seguimiento se anotará la cantidad de observaciones, de acuerdo con cada concepto, así como la cantidad de observaciones a las que se le dio seguimiento, a aquellas a las que se les dio solución y las que están siendo atendidas.</p> <p>Para auxiliarnos en los requerimientos de datos, su definición y la forma en que se capturan en el documento, la guía de auditoría y los ejemplos de los documentos (anexos a la minuta), son la base.</p> <p>(LG) y (AC) definen los datos redundantes y los nuevos en la captura del documento resultando:</p> <ul style="list-style-type: none"> ▪ Área Auditada (Viene de Orden de Auditoría) ▪ Clase de Auditoría (Viene de Orden de Auditoría) ▪ No. Progresivo de Revisión (Viene de Orden de Auditoría) ▪ Fecha de inicio y fecha de conclusión (Viene de Orden de Auditoría) ▪ Período revisado (Viene de Orden de Auditoría) ▪ Nombre de auditores responsables (Viene de Orden de Auditoría y es sólo uno) ▪ Cantidad de auditores Asignados (Viene de Orden de Auditoría), a la auditoría. ▪ Tiempo real empleado (Nuevo Dato) ▪ Tiempo de supervisión (Nuevo Dato) ▪ No se almacena en la base de datos

	<ul style="list-style-type: none"> ▪ Observación (Nuevo Dato) ▪ Seguimiento (Nuevo Dato) ▪ Rubro (Nuevo Dato) ▪ Cantidad (Nuevo Dato) ▪ Concepto (Nuevo Dato)
19	<p>(LG) explica a detalle el funcionamiento del módulo Evaluación de auditoría a (AC): La Evaluación de auditoría es el documento donde se registra la evaluación global de la auditoría. La evaluación se lleva a cabo en dos aspectos: el correcto y completo uso de todos los documentos de la auditoría (conforme a la guía de auditoría) y la correcta ejecución del proceso (de acuerdo a planes y tiempos determinados en la misma).</p> <p>Para auxiliarnos en los requerimientos de datos, su definición y la forma en que se capturan en el documento, la guía de auditoría y los ejemplos de los documentos (anexos a la minuta), son la base.</p> <p>(LG) y (AC) definen los datos redundantes y los nuevos en la captura del documento resultando:</p> <ul style="list-style-type: none"> ▪ Área auditada (Viene de orden de auditoría) ▪ Clase de auditoría (Viene de orden de auditoría) ▪ No de orden de auditoría (Viene de orden de auditoría) ▪ Fecha de inicio y fecha de conclusión (Viene de orden de auditoría) ▪ Período revisado (Viene de orden de Auditoría) ▪ Auditores responsables (Viene de Orden de Auditoría) <p>No se guardan en la base de datos</p> <ul style="list-style-type: none"> ▪ Evaluación (Nuevo Dato) ▪ Comentario (Nuevo Dato)
20	<p>Este módulo brindará la posibilidad de poder consultar la información de la base de datos del Sistema de Automatización del Proceso de Auditoría, que administra la información correspondiente a las observaciones de auditorías y seguimientos realizados.</p> <p>Se generarán cuatro tipos de reportes basados en los formatos proporcionados por el área:</p> <ul style="list-style-type: none"> ▪ <i>Resumen de órdenes de auditoría.</i>- Este reporte permite ver los datos más relevantes de una auditoría así como el número de observaciones que se determinaron en ellas y cuantas de las mismas resultaron no solventadas. ▪ <i>Detalle de observaciones.</i>- Con este reporte se obtienen por auditoría las observaciones que se generaron en la misma (número de observación, título, recomendación correctiva y preventiva, estatus, fecha de implementación y fecha de solventación). ▪ <i>Resumen de observaciones de auditoría.</i>- Consiste en un seguimiento de las observaciones desde la auditoría en que fueron creadas y los seguimientos que han recibido, además de indicar si fueron atendidas. ▪ <i>Total de observaciones.</i>- Presenta la cuenta de las observaciones generadas, las pendientes de seguimiento, las solventadas, las no solventadas y el total de observaciones pendientes por año. <p>Se permitirá filtrado de los reportes anteriores por:</p> <ul style="list-style-type: none"> ▪ No. de orden de auditoría ▪ No. Progresivo de revisión ▪ No. oficio (informe)

	<ul style="list-style-type: none">▪ Fecha de inicio de la auditoría▪ Área▪ Tipo de auditoría▪ Observaciones relevantes/ No relevantes▪ Observaciones solventadas/ No solventadas▪ Clasificación de las observaciones▪ Se permitirá observar una vista previa del reporte elegido.▪ Se imprimirá el reporte elegido.▪ Se limpiarán los campos de búsqueda del reporte.
--	---

3.3.2 Definición del Sistema y Módulos a Desarrollar

Teniendo en cuenta los objetivos planteados anteriormente, se puede definir “SAPA” en diferentes módulos enfocados a la generación de cada uno de los informes y reportes de los que consta el proceso de auditoría llevado a cabo por la contraloría interna de cualquier institución de gobierno.

Es aquí donde se definen los datos que se capturan de cada formato. Algunos de ellos son datos fijos en los mismos, otros se almacenarán en la base de datos para ser empleados por otros formatos más adelante, y otros sólo se capturan en los formatos porque su información no es relevante y por lo tanto no se almacena.

A continuación se muestra una descripción de los módulos que contendrá el sistema en base a las entrevistas realizadas:

3.3.2.1 Módulo Programa Anual de Control y Auditoría (PACA)

- Permitir la captura, edición y consulta del PACA Resumen y PACA Detallado de manera organizada.
- La organización de la información debe presentarse en la forma en que se encuentra organizada en los documentos (Un título contiene varios rubros y un rubro contiene varias descripciones)
- Filtrar la información a consultar por año de PACA.
- Generar los documentos de PACA Resumen y PACA Detallado en Excel de manera automática.

3.3.2.2 Inicio de Auditoría

3.3.2.2.1 Módulo orden de auditoría

- Cargar números progresivos de revisión para su selección desde el “PACA”.
- Cargar información de áreas, tipos de auditorías, tipo de personal y personal de catálogos.
- Permitir la captura, edición y consulta de la información correspondiente a la orden de auditoría a capturar.
- Generación automática y consecutiva del número de orden de auditoría, en base al año y un número consecutivo.
- Elección del tipo de auditoría: nueva o seguimiento.
- Validación y carga de personal a quien se dirige la atención de la orden, responsables de la misma y personal a quien se dirigen las copias.
- Validación y uso de módulo de “**Consulta y selección de observaciones (inventario)**”, dependiendo del tipo de auditoría seleccionada (solamente para auditorías de seguimiento).
- Validar que los auditores sólo puedan consultar la información de órdenes de auditoría de las auditorías en las que sean responsables.
- Generar órdenes de auditoría en “Word” de manera automática, realizando las consultas de información necesaria desde otros módulos.

3.3.2.2.2 Módulo de consulta y selección de observaciones (inventario)

- Consulta de observaciones con los siguientes campos:
 - No. orden de auditoría original
 - No. progresivo de revisión
 - No. de observación
 - Avance
 - Observación original
 - Situación actual
 - Recomendación correctiva
 - Recomendación preventiva
- Filtrar las observaciones a consultar por área auditada y clase (tipo) de auditoría a la que pertenece la observación.
- Consulta de la última orden de auditoría que dio seguimiento a la observación que se consulta.
- Este módulo será empleado por las órdenes de auditoría de seguimiento, por lo que debe de ser llamado desde el módulo de “**Orden de Auditoría**” para el caso de auditorías de seguimiento.
- Permitir la selección de las observaciones para integrarlas a una orden de auditoría de seguimiento.

3.3.2.2.3 Módulo Acta de Inicio

- Carga de la información de catálogos de personal y tipo de personal.
- Consulta de datos generales de auditoría en curso (Clase, No. de Orden, No. progresivo de revisión)
- Permitir captura, edición y consulta de información de acta de inicio con base en la auditoría a la que pertenece.
- Validar y cargar servidor auditado y testigos del catálogo de tipo de personal.
- Validar que los auditores sólo puedan consultar la información de actas de inicio de las auditorías en las que sean responsables.
- Generar actas de inicio en “Word” de manera automática, realizando las consultas de información necesaria desde otros módulos.

3.3.2.2.4 Módulo carta planeación

- Consulta de los datos generales de la auditoría en curso (Clase, No. de orden, No. progresivo de revisión).
- Permitir captura, edición y consulta de información de carta planeación con base en la auditoría a la que pertenece.
- Validar que los auditores sólo puedan consultar la información de cartas de planeación de las auditorías en las que sean responsables.
- Generar carta planeación en “Word” de manera automática, realizando las consultas de información necesaria desde otros módulos.

3.3.2.2.5 Módulo cronograma de actividades

- Generar cronogramas de actividades en “Word” de manera automática, realizando las consultas de información necesaria desde otros módulos. (Sólo se llenará el encabezado del formato con datos de otros módulos pues la información que contiene no es importante almacenarla)

3.3.2.3 Ejecución de Auditoría

3.3.2.3.1 Módulo marco conceptual

- Consulta de los datos generales de la auditoría en curso (Clase, No. de orden, No. progresivo de revisión).
- Permitir captura, edición y consulta de diferentes cédulas de marcos conceptuales con base en la auditoría a la que pertenecen y su número de cédula. (Un marco contiene varias cédulas).
- Validar que los auditores sólo puedan consultar la información de marcos conceptuales de las auditorías en las que sean responsables.

- Generar marco conceptual en “Word” de manera automática, realizando las consultas de información necesaria desde otros módulos.

3.3.2.3.2 Módulo cédulas de observaciones

- Consulta de los datos generales de la auditoría en curso (Clase, No. de orden, No. progresivo de revisión).
- Permitir captura, edición y consulta de diferentes cédulas de observaciones con base en la auditoría a la que pertenecen y su número de observación. (Una orden de auditoría contiene varias observaciones).
- Al generar las observaciones, debe existir un proceso por medio del cual se validen aquellas que son aprobadas por el Director de Supervisión y Auditoría, y discutidas con el área. Únicamente las que sean aprobadas se tomarán como definitivas, pasarán al inventario de observaciones y su número de observación se actualizará en función del número de observaciones que resulten (las observaciones no aprobadas deberán ser borradas del sistema).
- Validar que los auditores sólo puedan consultar la información de las cédulas de observaciones de las auditorías en las que sean responsables.
- Generar cédula de observaciones en “Word” de manera automática, realizando las consultas de información necesaria desde otros módulos.

3.3.2.3.3 Módulo cédulas de seguimientos

- Consulta de los datos generales de la auditoría en curso (Clase, No. de orden, No. progresivo de revisión).
- Permitir captura, edición y consulta de diferentes cédulas de seguimientos con base en la auditoría a la que pertenecen y su número de observación (Un orden de auditoría seguimiento contiene varias observaciones).
- Consultar cuál es la última auditoría que dio seguimiento a la observación que se esta consultando.
- Validar que al llegar alguna observación a un avance en el seguimiento del 100% y sea definida como solventada salga del inventario de observaciones y no pueda ya ser modificada.
- Permitir que las observaciones sean desligadas de la orden de auditoría de seguimiento si el usuario así lo requiere.
- Generar un procedimiento que permita realizar nuevos replanteamientos (nuevas recomendaciones preventivas y correctivas) a las observaciones.
- Validar que los auditores sólo puedan consultar la información de las cédulas de seguimientos de las auditorías en las que sean responsables.
- Generar cédula de seguimientos en “Word” de manera automática, realizando las consultas de información necesaria desde otros módulos.

3.3.2.4 Informe de Auditoría

3.3.2.4.1 Módulo informe de auditoría

- Carga de la información de los catálogos de personal y tipo de personal.
- Consulta de los datos generales de la auditoría en curso (Clase, No. de orden, No. progresivo de revisión).
- Permitir captura, edición y consulta de información del informe de auditoría con base en la auditoría a la que pertenece.
- Validar y cargar copias de informe del catálogo de tipo de personal.
- Validar que los auditores solo puedan consultar la información de los informes de auditoría de las auditorías en las que sean responsables.
- Generar informes de auditoría en “Word” de manera automática, realizando las consultas de información necesaria desde otros módulos.

3.3.2.4.2 Módulo Cédula única de auditoría

- Consulta de los datos generales de la auditoría en curso (Clase, No. de orden, No. progresivo de revisión)
- Permitir captura, edición y consulta de información de cédulas únicas de auditoría con base en la orden de auditoría a la que pertenece.
- Validar que los auditores sólo puedan consultar la información de cédulas únicas de las auditorías en las que sean responsables.
- Generar cédulas únicas en “Word” de manera automática, realizando las consultas de información necesaria desde otros módulos.

3.3.2.4.3 Módulo evaluación de auditoría

- Generar Evaluación de Auditoría en “Word” de manera automática, realizando las consultas de información necesaria desde otros módulos. (Solo se llenará el encabezado del formato con datos de otros módulos pues la información que contiene no es importante almacenarla).

3.3.2.5 Módulo Catálogos

- Este módulo contendrá un listado completo de los catálogos que se usarán dentro del sistema permitiendo llevar un control de la creación, modificación y eliminación de datos, dando de alta o baja los registros según sea el caso.
- El sistema contará con tres diferentes catálogos:
 - CLASES DE AUDITORÍAS.- Son los nombres de las Auditorías que serán llevadas a cabo y se dividen de la siguiente manera:

1. *Integrales*
 - 1.1. *Reporte Financiero*
 - 1.1.1. *Sistema de Información Registro*
 - 1.2. *Tesorería*
 - 1.2.1. *Disponibilidades*
 - 1.3. *Egresos*
 - 1.3.1. *Adquisiciones, arrendamiento y servicios*
 - 1.3.2. *Pasivos*
 - 1.3.3. *Presupuesto - Gasto Corriente*
 - 1.4. *Nóminas*
 - 1.4.1. *Recursos Humanos*
 - 1.5. *Conversión*
 - 1.5.1. *Inventarios y activos fijos*
 - 1.6. *Presupuesto- Gasto de inversión*
2. *Específicas*
 - 2.1. *Sistema de Información y Registro*
 - 2.1.1. *Sistemas Automatizados*
 - 2.2. *Sistema Integral de información (SII)*
3. *Evaluación de programas*
4. *Del Desempeño*
 - 4.1. *Dirección General Jurídica*
 - 4.2. *Dirección General de Planeación Financiera*
 - 4.3. *Dirección General de Inspección*
 - 4.4. *Dirección General de Vigilancia*
 - 4.5. *Dirección General de Comunicación y Vinculación*
 - 4.6. *Dirección de Organización de Gobierno y Comités Interinstitucionales*
 - 4.7. *Dirección General de Riesgos*
 - 4.8. *Dirección General de Sanciones*
5. *Seguimientos. A todas las anteriores*
 - **ÁREAS.-** Serán los nombres de las áreas a las que les será realizada algún tipo de Auditoría.
 - **PERSONAL.-** Serán los nombres de usuarios que estarán realizando alguna actividad sobre el sistema así como sus datos personales.

3.3.2.6 Módulo Reportes

Este módulo brindará la posibilidad de poder consultar información de la base de datos “SAPA”, que administra la información correspondiente a las observaciones de auditorías y seguimientos realizados; además de poder imprimir los reportes generados de dichas consultas.

Genera cuatro tipos de reportes basándose en los formatos proporcionados por el área:

- Resumen de órdenes de auditoría.
- Detalle de observaciones.
- Resumen de observaciones de auditoría.
- Total de observaciones.
-

Permite filtrado de los reportes anteriores por:

- No. de orden de auditoría.
- No. progresivo de revisión.
- No. oficio (informe).
- Fecha de inicio de la auditoría.
- Área.
- Tipo de auditoría.
- Observaciones relevantes/ no relevantes.
- Observaciones solventadas/ no solventadas.

Clasificación de las observaciones.

Permitir observar una vista previa del reporte elegido.

Imprimir el reporte elegido.

Limpiar los campos de búsqueda del reporte.

CAPÍTULO 4



Análisis y Diseño

4.1 Análisis

4.1.1 Análisis de Requerimientos del Sistema

Para el desarrollo del sistema y su posterior operación se eligió como sistema operativo *Windows 2000* por los siguientes motivos:

- Sistema operativo empleado en oficinas de contraloría.
- Conocimiento de los usuarios de este sistema operativo y de sus aplicaciones (en especial *Microsoft Office*, aplicación en la cuál desarrollan la mayor parte de la documentación del proceso de auditoría).
- Facilidad para la instalación y manejo de las herramientas comerciales de desarrollo de aplicaciones visuales (*Delphi*, *Visual Basic*, *Fox Pro*, entre otras).

Las otras dos opciones comerciales, UNIX en cualquiera de sus versiones (*Solaris*, *Linux*, etc.) o *Mac'os* de *Macintosh* son sistemas operativos que no se emplean en oficinas de contraloría y los usuarios están poco familiarizados con su uso.

Se eligió *Windows 2000 Server* como sistema operativo para el servidor debido a que es un sistema operativo que:

- Permite la administración en una red LAN tipo estrella con protocolo Ethernet
- Permite la instalación y manejo de un servidor de base de datos necesario para la operación del sistema (SQL Server).
- De los sistemas operativos de red, es la opción más económica (a excepción de *Linux*).

Se eligió *Delphi 5.0* porque:

- Permite el desarrollo de aplicaciones visuales para plataforma *Windows*.
- Es completamente compatible con el *DBMS SQL Server 7.0*.
- El manejo de los objetos de conexión a la base de datos es más sencillo que en otros Lenguajes Visuales.
- El manejo de objetos es más confiable y completo que en otros lenguajes para desarrollo de aplicaciones visuales.
- Es un lenguaje orientado a objetos.

Se eligió *Office* para la generación automática de los documentos porque es el paquete con el que se elaboran los documentos oficiales en el proceso de auditoría.

Se eligió una Red *LAN Ethernet* a 100 Mbps con topología de estrella porque:

- Es una topología de red confiable pues cada cliente opera de manera independiente.
- Permite un mejor aprovechamiento de la velocidad de la red de manera adecuada al evitar el congestionamiento producido en el tipo Anillo y Bus.
- Permite la operación de diversos clientes sobre la misma base de datos instalada en el servidor (Manejo de la información centralizada), es decir, el uso de la arquitectura Cliente-Servidor.
- Evita el empleo de espacio de almacenamiento de datos innecesario en los clientes al instalar las aplicaciones y la base de datos en el servidor y no en los clientes.
- Evita carga de proceso de la información en los clientes pues las consultas se pueden procesar en el servidor.

Se eligió *SQL Server* porque:

- Es un DBMS económico (Comparado con *SYBASE*, *Informix* y *Oracle*).
- Trabaja sobre *Windows 2000*.

- Permite la operación de diversos clientes sobre la misma base de datos instalada en el servidor (Manejo de la información centralizada).
- El manejo de bases de datos es confiable.
- Permite el manejo adecuado de la integridad referencial y la implementación de modelos relacionales de bases de datos por medio de *triggers*, *constraints*, llaves, etc.
- Permite el acceso de clientes en red por medio de ODBC o de manera directa dependiendo de la tecnología.
- Permite el desarrollo de consultas que se almacenan y ejecutan en el servidor (*Stored Procedures*), lo que permite quitar trabajo de proceso de datos a los clientes.
- Es compatible con el *DBEngine* de *Delphi*.

En la siguiente figura se muestra una tabla que define en qué parte del proyecto se emplearon las elecciones anteriores:

Actividad	Desarrollo del sistema	Operación del sistema en los clientes	Operación del sistema en el servidor
Sistema Operativo	Windows 2000	Windows 2000	Windows 2000 Server
Software	Delphi 5.0	Office 2000	
Red	Velocidad 100 Mbps Protocolo Ethernet Topología Estrella	Velocidad 100 Mbps Protocolo Ethernet Topología Estrella	Velocidad 100 Mbps Protocolo Ethernet Topología Estrella
DBMS	SQL Server 7.0	SQL Server 7.0	SQL Server 7.0
Hardware	<ul style="list-style-type: none"> • Computadoras personales • Pentium III a 900 Mhz • Memoria RAM: 256 MB • Espacio disco duro: 2 GB • Monitor de 15" al menos 	<ul style="list-style-type: none"> • Computadoras personales • Pentium III a 900 Mhz • 256 MB en memoria RAM • Espacio disco duro: 500 MB • Monitor de 15" al menos • Mouse • Tarjeta de red Ethernet 	Compaq Proliant ML 350 <ul style="list-style-type: none"> • Xeon a 3 GHz • 1 GB en memoria RAM • 4 discos duros de 40GB • Unidad de cinta dat 8

Tabla 4.1 Recursos materiales

En la selección de los recursos humanos a emplear en el desarrollo de “SAPA” se definieron los siguientes:

#	Roles propuestos para el proyecto	Participación en el proyecto
1	Un Líder de proyecto	11 meses
2	Un analista programador	11 meses

Tabla 4.2 Recursos humanos

Lo óptimo hubiese sido que al menos un programador más participara en el desarrollo del proyecto para disminuir las cargas de trabajo e incrementar la holgura de los tiempos de desarrollo del proyecto.

4.1.2 Planeación de Actividades

El análisis y el diseño de sistemas involucra actividades de naturaleza muy diferente que al integrarse constituyen un proyecto. Para que un proyecto tenga éxito, será importante administrar de manera adecuada las actividades que lo componen a través de la planeación y el control.

La planeación consiste en la organización de las siguientes actividades:

- Selección del personal que participa en el proyecto (equipo de trabajo).
- Asignación de tareas adecuadas a cada miembro del equipo de trabajo.
- La estimación del tiempo que requiere cada tarea para su ejecución.
- Programación o calendarización de las tareas.

Cada una de las actividades anteriores puede ser subdividida para obtener una mejor descripción y organización.

El control consiste en darle seguimiento a las actividades determinadas durante la planeación de la siguiente manera:

- Comparación y evaluación de avances del proyecto con respecto del plan.
- Implementar medidas para acelerar y corregir las desviaciones.
- Reprogramar las actividades que sean necesarias.
- Motivación del equipo de trabajo.

La tarea de llevar a cabo una adecuada planeación de actividades implica, primero, determinar el total de actividades que deberán llevarse a cabo, después, definir el tiempo que tomará realizar cada una de ellas, la fecha en que deben llevarse a cabo y el orden que deberán seguir para que el proyecto se concluya.

En el caso de “*SAPA*”, la definición de las actividades a realizarse se llevó a cabo con base en el desglosamiento en subactividades de las fases del modelo clásico de desarrollo de sistemas: estudio inicial, análisis, diseño, desarrollo, pruebas e implantación.

Basándose en la experiencia anterior, se llevaron a cabo las estimaciones de la duración de cada una de estas subactividades. Se pueden hacer estimaciones optimistas y pesimistas para después obtener un promedio de duración de las actividades, pero esto no es lo más recomendable, en todo caso, será mejor detallar lo más posible las actividades a desarrollar en un proyecto para evitar un alto o mediano grado de incertidumbre y evitar las sorpresas.

Para construir el plan de “*SAPA*” se empleó un diagrama de Gantt bidimensional que es una manera sencilla de programar las actividades que se llevan a cabo de manera simultánea en un proyecto. Su definición es sencilla: diagrama en el que las actividades son representadas por barras. La longitud de dichas barras es la duración de cada actividad.

A continuación se muestra el diagrama de Gantt de “*SAPA*”:

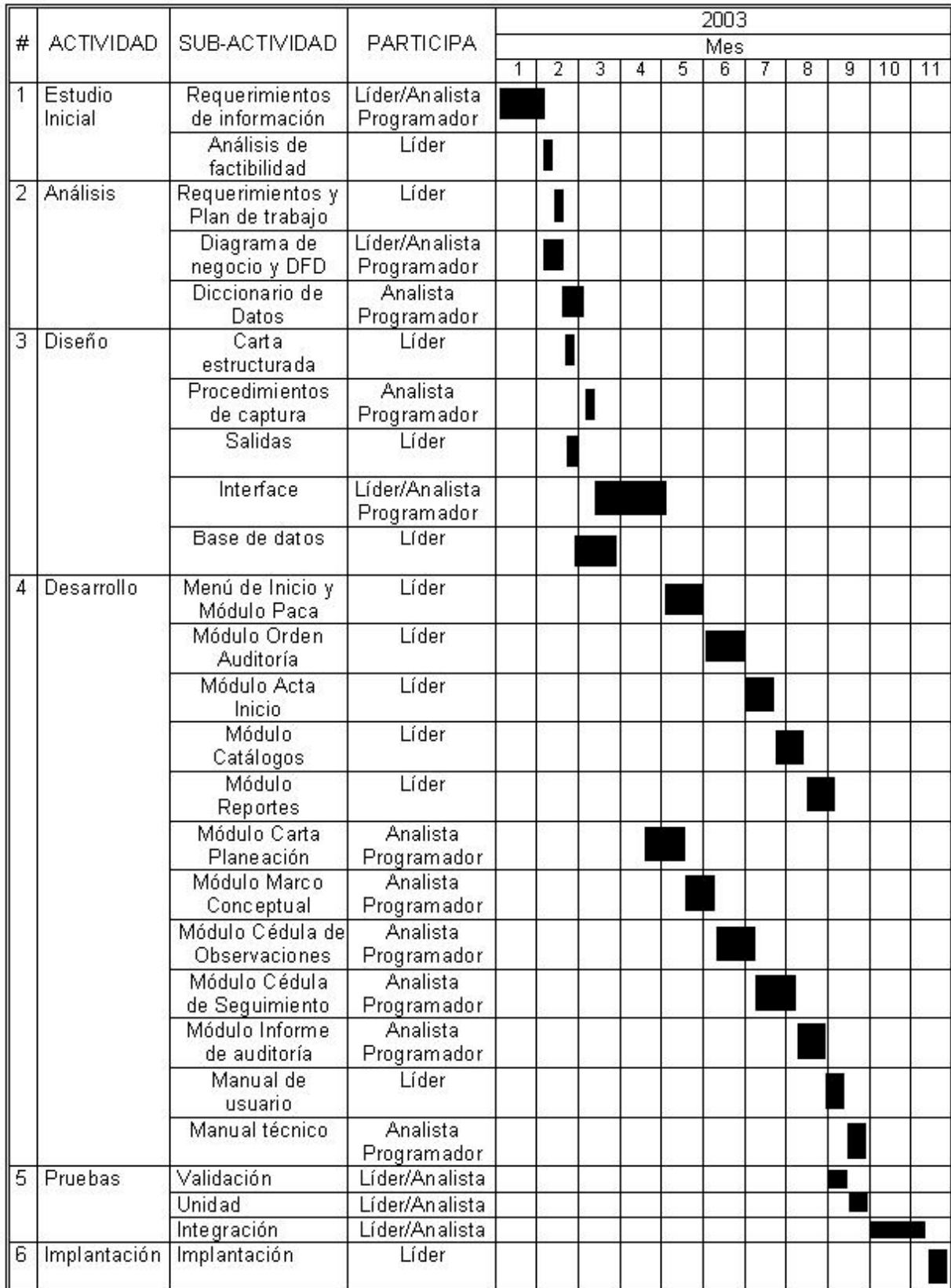


Figura 4.1 Diagrama de Gantt de “SAPA”

4.1.3 Diagrama de Flujo de Datos (DFD)

Es una técnica que representa gráficamente la forma en que los datos se desplazan entre las entradas y salidas del sistema, así como los procesos que los modifican. El DFD permite esquematizar un sistema en diferentes niveles de abstracción y es posible dividirlo de tal forma que cada nivel contenga un mayor flujo de información y un mayor detalle funcional.

Un DFD de nivel 0, también es denominado modelado fundamental del sistema y representa al sistema completo como una sola burbuja con datos de entrada y salida. Al particionar al DFD de nivel 0 para mostrar más detalles, aparecen representados procesos y flujos de información adicionales. A medida que se detalla el DFD se observa en forma más precisa como funcionará el sistema.

La notación del DFD identifica a las entidades externas (hardware, personas, otros programas) que producen o reciben información como rectángulos dobles (con sombra); un rectángulo con bordes redondeados representa un proceso o una transformación de los datos; un almacenamiento de datos toma la forma de un rectángulo más delgado y el flujo de datos es representado con flechas etiquetadas. A continuación se muestran los elementos de esta notación:

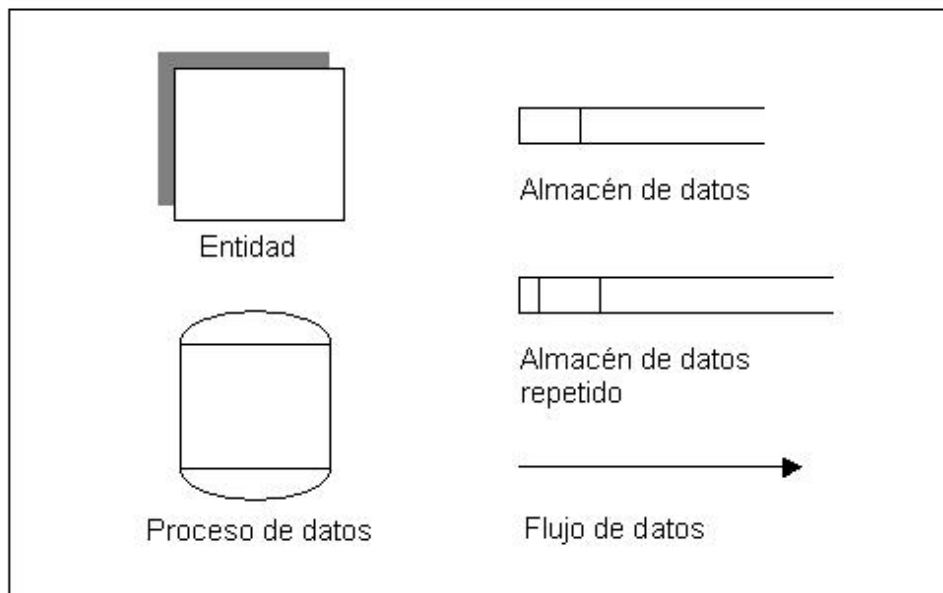
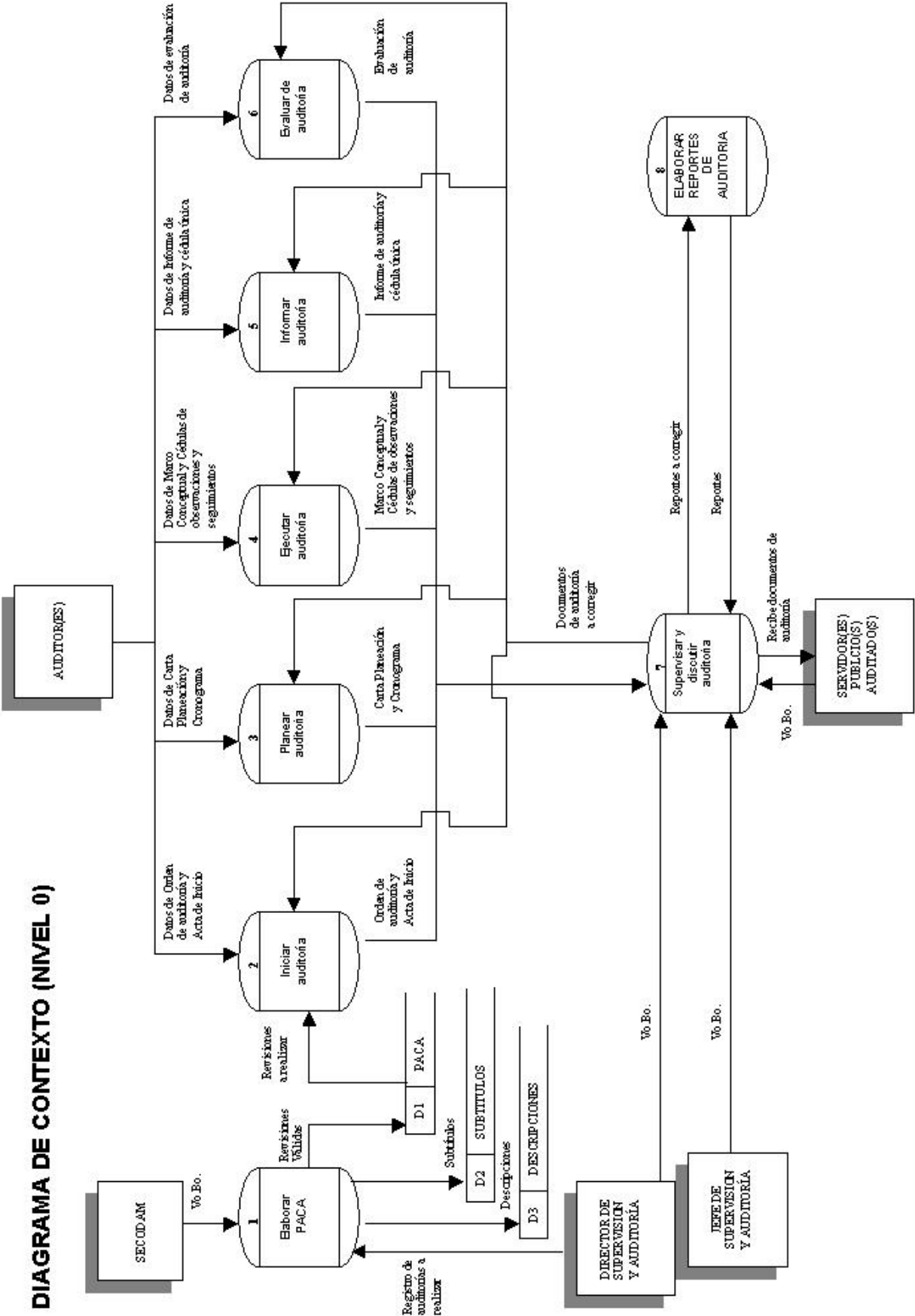
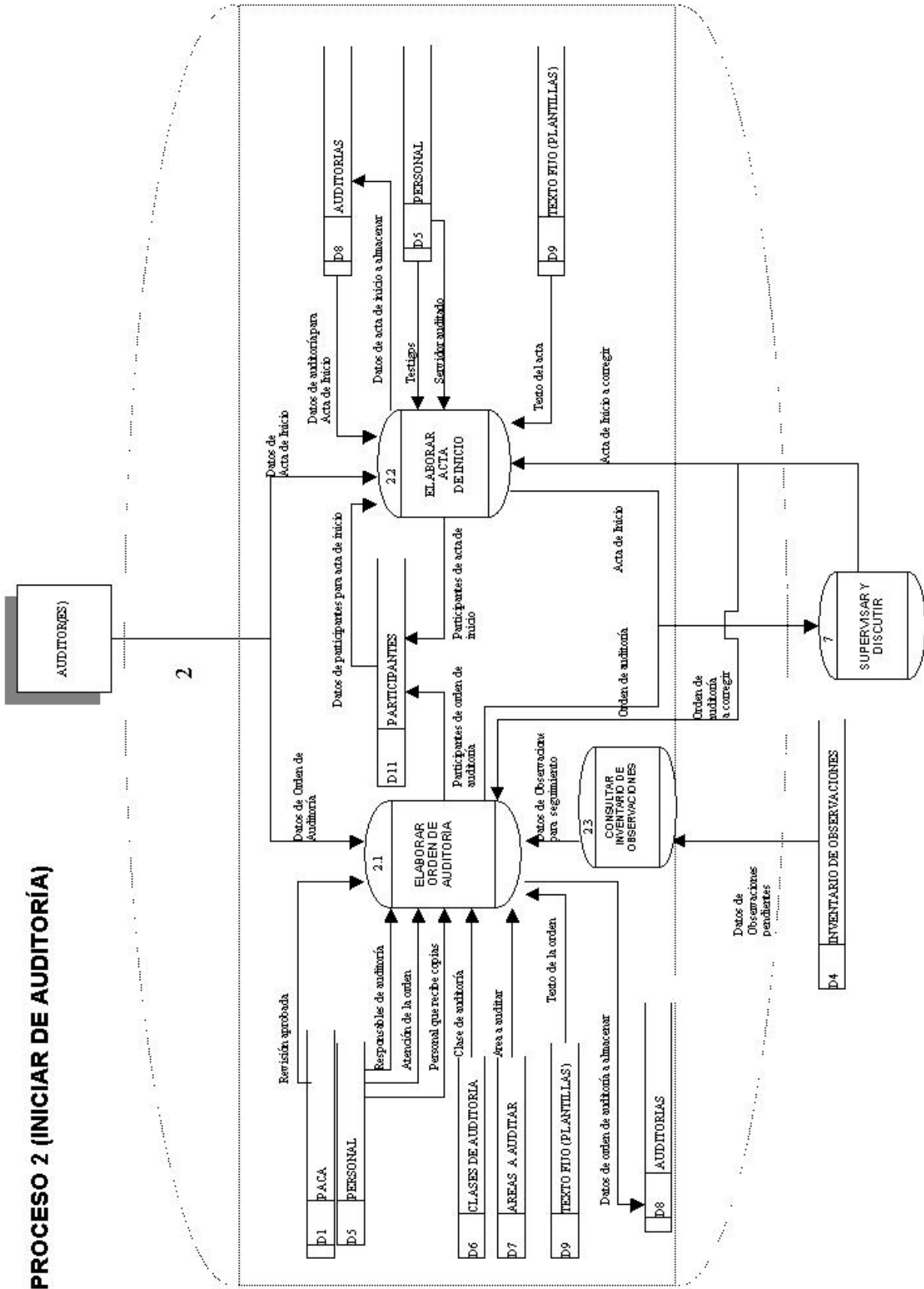


Figura 4.2 Elementos de un DFD

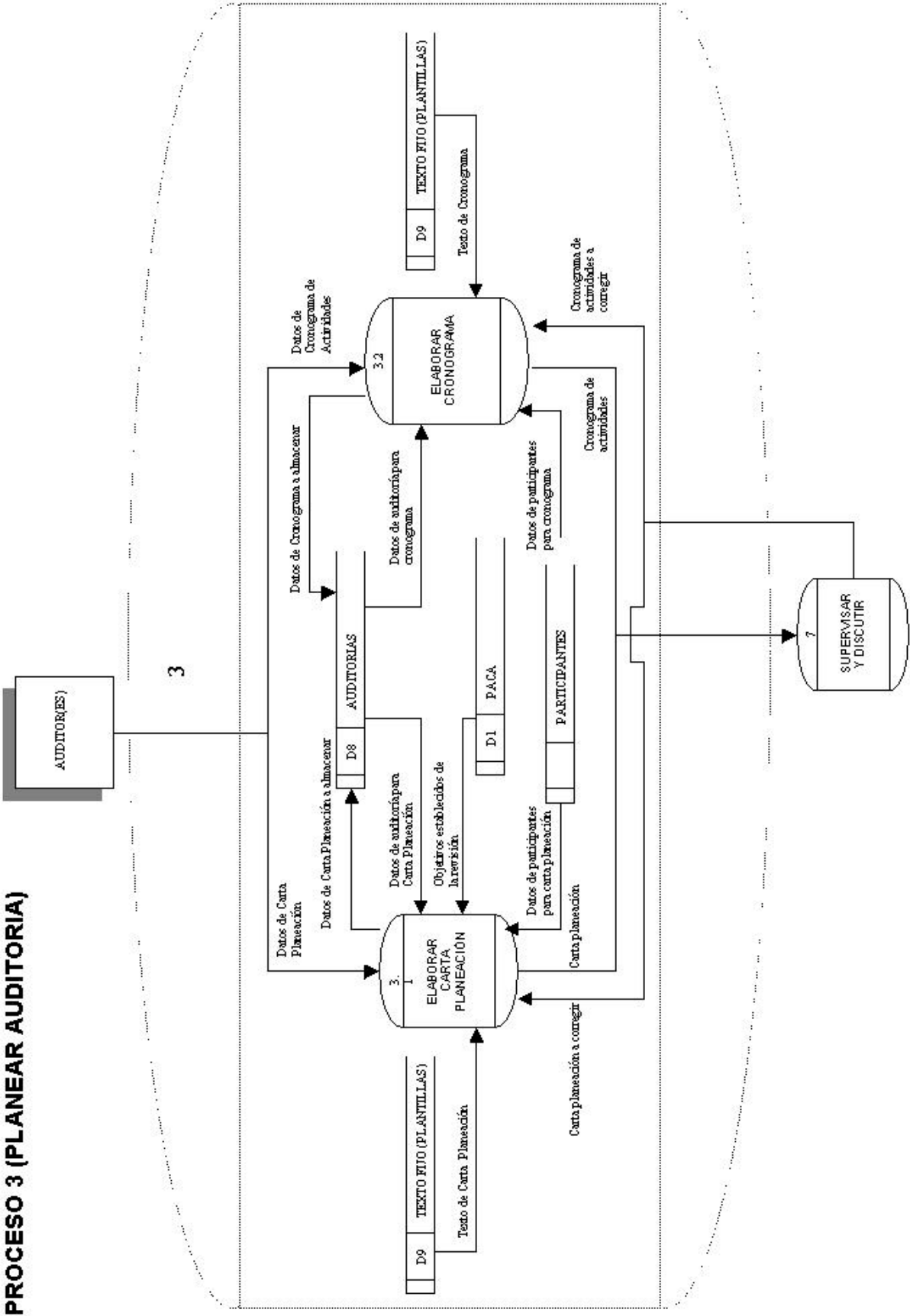
En el desarrollo de “SAPA” sólo fue necesario diseñar el diagrama de nivel 0 ó contexto y explotar los procesos que contienen un nivel más.

DIAGRAMA DE CONTEXTO (NIVEL 0)

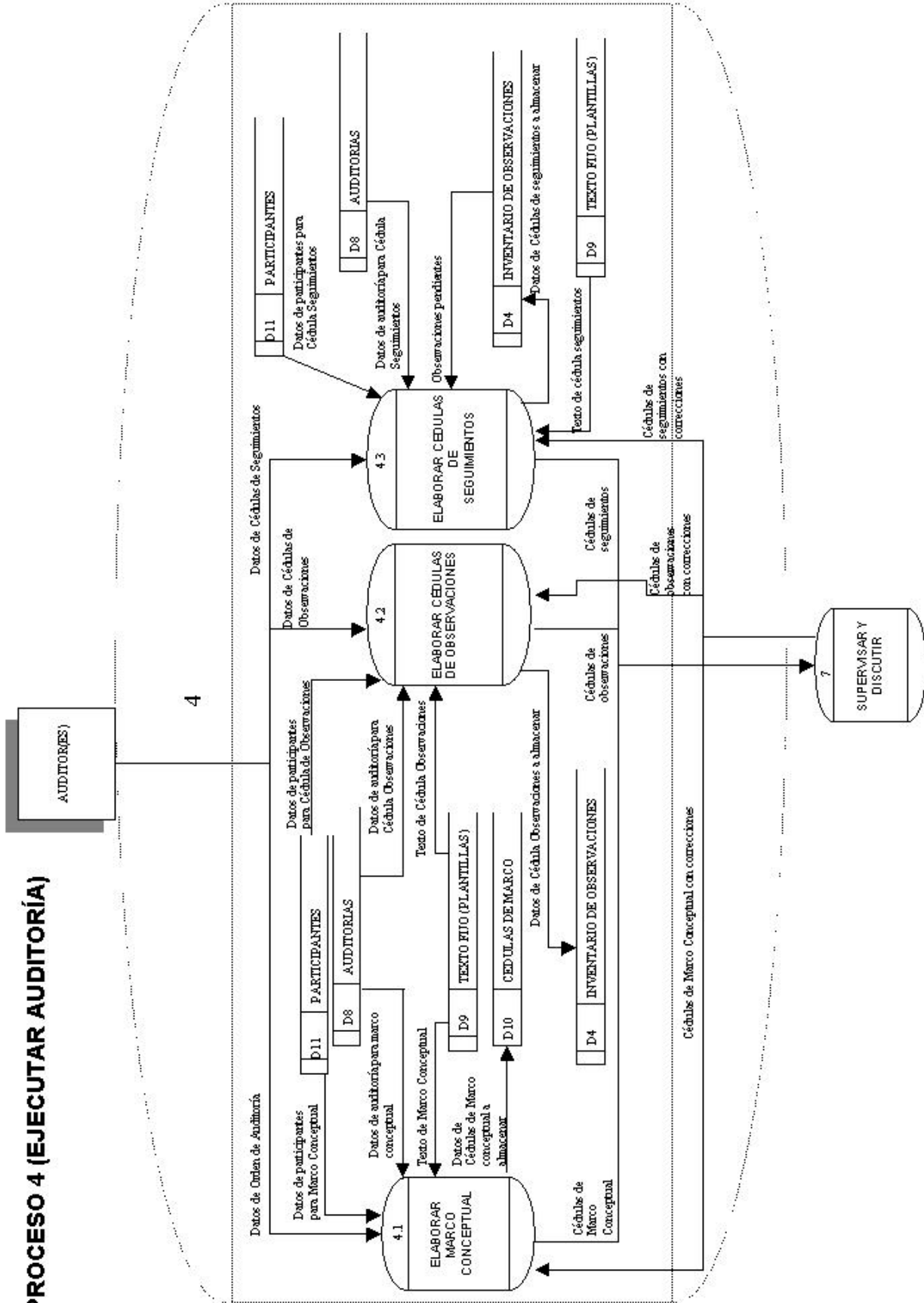




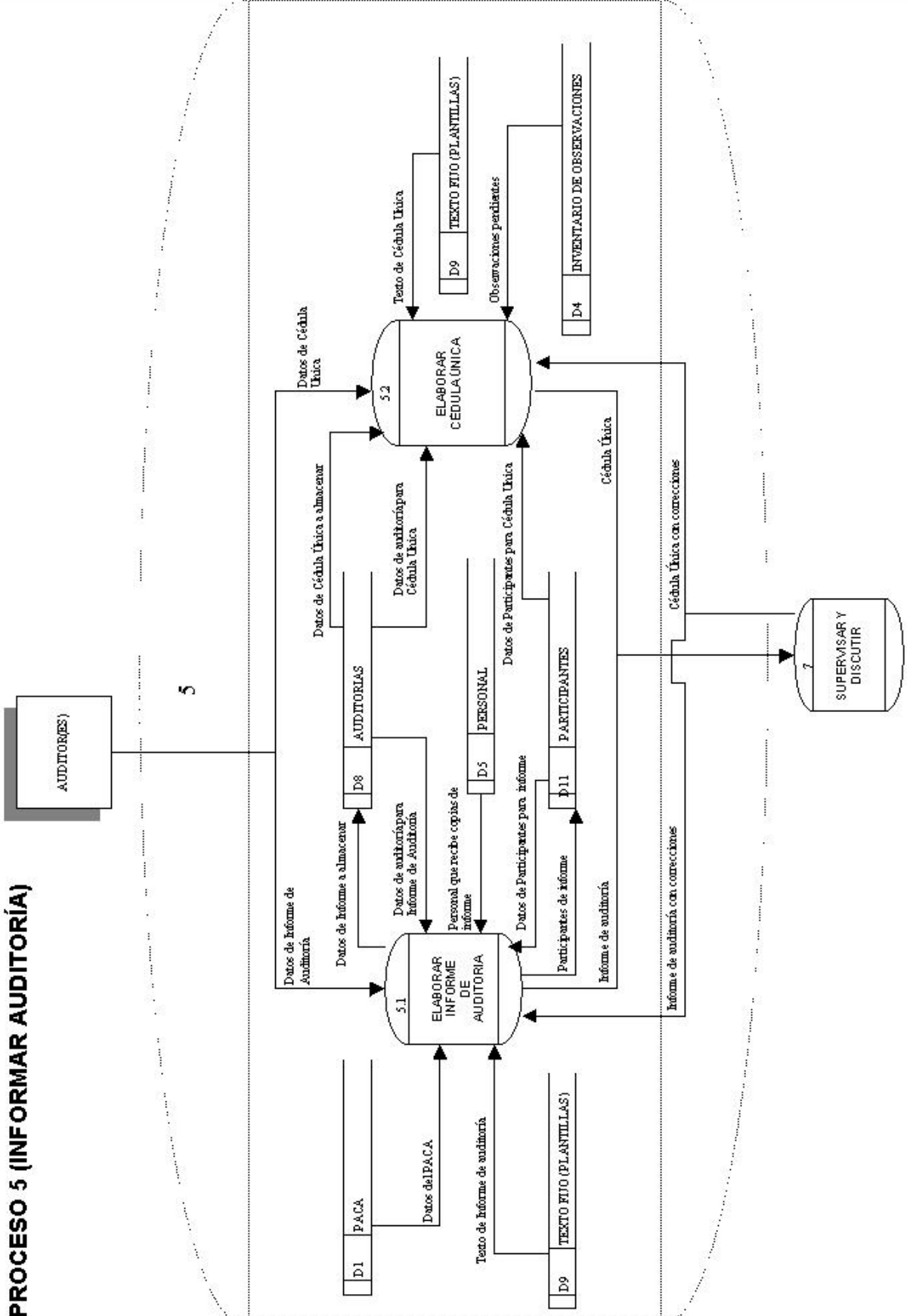
PROCESO 3 (PLANEAR AUDITORIA)



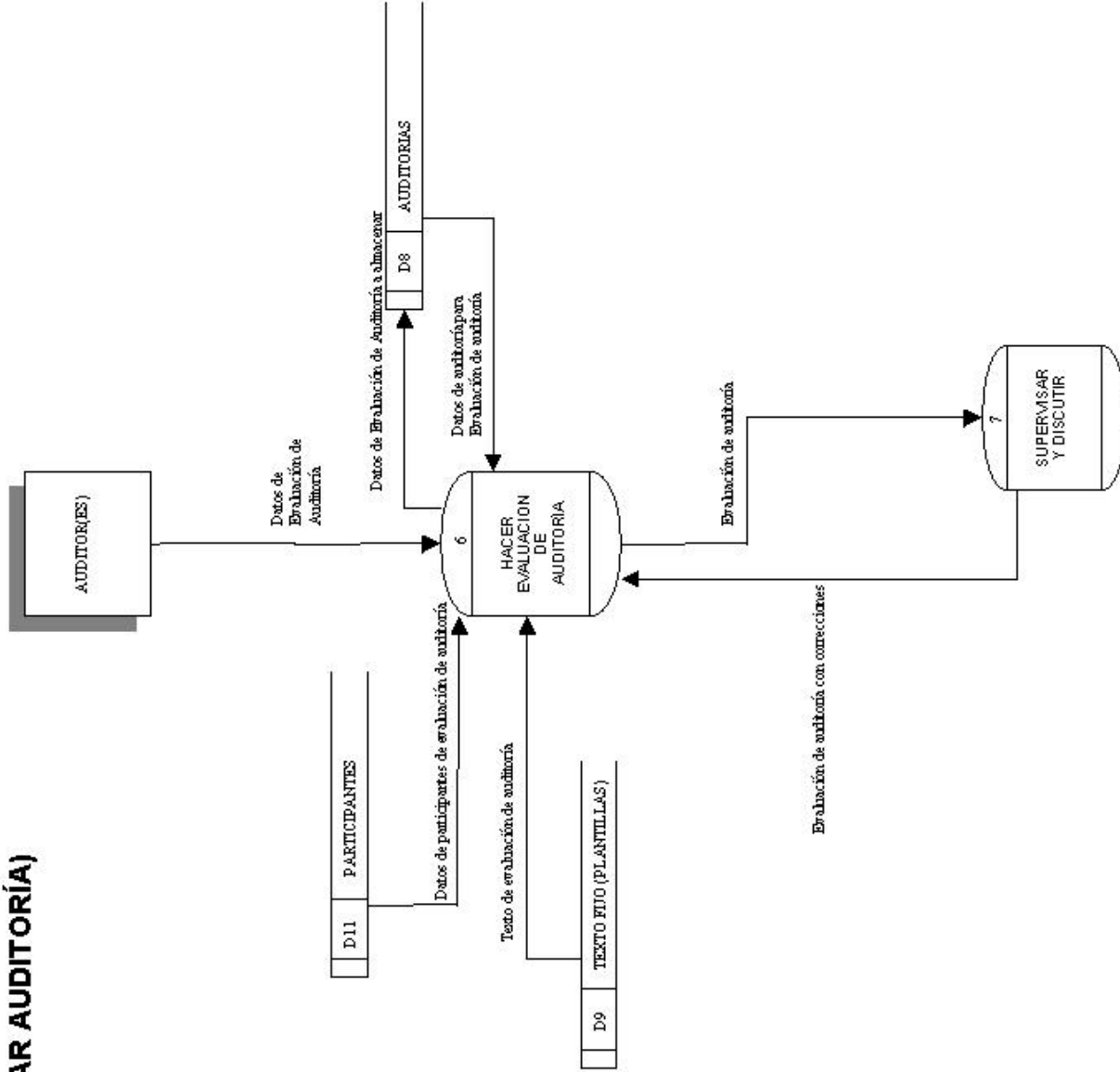
PROCESO 4 (EJECUTAR AUDITORÍA)



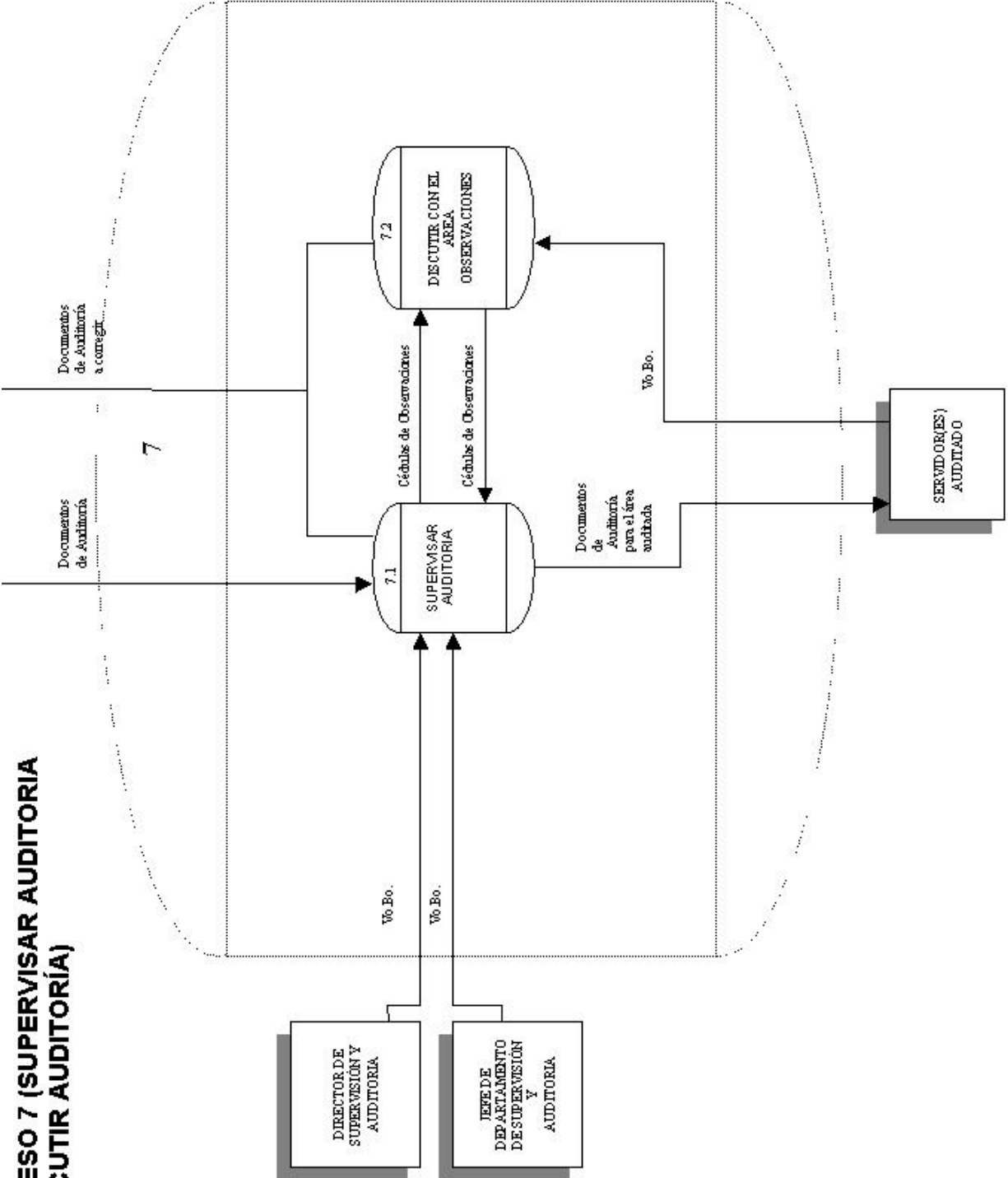
PROCESO 5 (INFORMAR AUDITORÍA)



PROCESO 6 (EVALUAR AUDITORÍA)

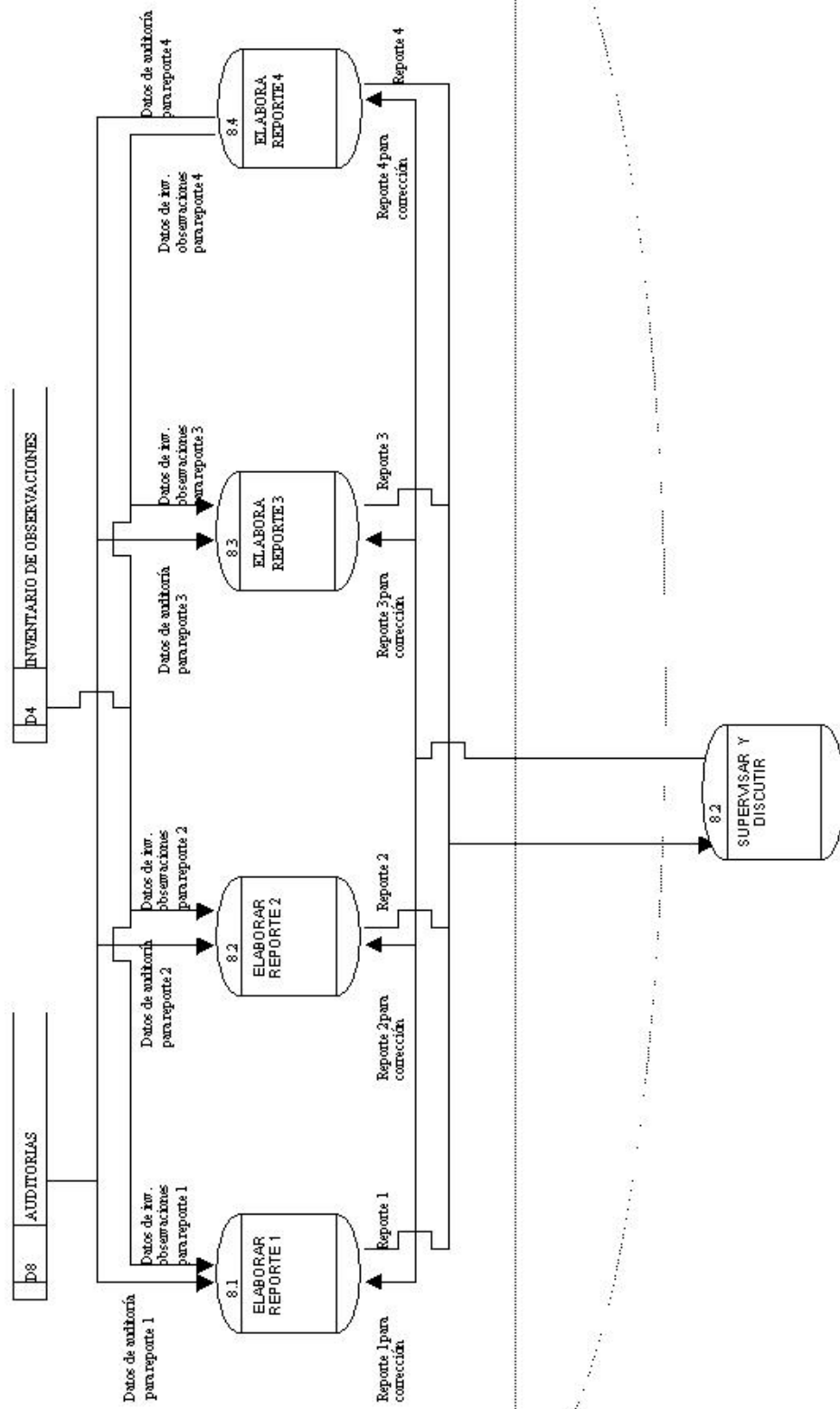


PROCESO 7 (SUPERVISAR AUDITORIA Y DISCUTIR AUDITORÍA)



PROCESO 8 (REPORTES)

8



4.1.4 Construcción del Diccionario de Datos

4.1.4.1 Definición

El diccionario de datos es una referencia de los datos y sus atributos recopilados por el analista de sistemas, para ser empleados durante el proceso de desarrollo de un sistema. Una de sus funciones más importantes es la de servir como principal traductor de términos empleados por la gente de la organización y proporcionar una base confiable para el modelado de la base de datos.

La base para la creación del diccionario de datos es el diagrama de flujo de datos (DFD) revisado en la sección anterior.

Otra de las funciones del diccionario de datos es la de catalogar los términos duplicados, es decir, pueden existir en un flujo de datos o en un almacén de datos del DFD, dos términos que hagan referencia al mismo dato. Al construir el diccionario, se recopilan todos los términos provenientes de flujos de datos y de almacenes para realizar una comparación entre ellos (cruce) y obtener aquellos que hacen referencia al mismo dato y por lo tanto son información redundante. El análisis mencionado permite obtener un mayor aprovechamiento de la información manejada en una organización favoreciendo entre otras cosas la comunicación entre diferentes personas, departamentos, áreas, etc. Un diccionario de datos también puede servir como la estandarización de los datos manejados en una organización.

Una visión general de los atributos que deberá contener un diccionario de datos es la siguiente:

- Nombre del dato.- Es la manera de identificar un dato, de la manera más sencilla y general posible.
- Descripción.- Se debe de incluir una definición concisa, pero informativa para que cualquiera la consulte.
- Longitud.- Número de caracteres o símbolos permitidos para un dato.
- Rango permitido.- Valores posibles que puede tomar un dato, por ejemplo, el rango para número de meses en el año es de 1 a 12.

4.1.4.2 Construcción

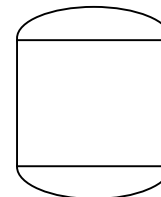
Para la construcción del diccionario de datos de “SAPA” se empleó la siguiente metodología:

- Se documentaron:
 - a) Procesos del DFD con flujos de datos entrantes y salientes.
 - b) Flujos de datos incluyendo las estructuras de datos que viajan en cada uno de ellos.
 - c) Almacenes de datos, incluyendo las estructuras de datos que contiene cada uno de ellos.
 - d) Documentación de las estructuras de datos incluyendo datos elementales sin detalle.

- Comparación de las estructuras de datos y sus datos elementales para eliminación de redundancias.
- Se documentaron finalmente todos los datos elementales resultantes del paso anterior resultando el diccionario de datos.

A continuación se muestran partes de la documentación realizada para la construcción del Diccionario de Datos y el diccionario mismo:

4.1.4.2.1 Procesos



Proceso	1	
Nombre	Elaborar PACA	
Entradas	Descripción del proceso	Salidas
Vo.Bo. de SECODAM Registro de auditorías a realizar	La Dirección de Supervisión y Auditoría realiza el documento con el cuál se describen las auditorías que se llevarán a cabo durante un año, se envía a SECODAM y se recibe su aprobación. Los datos son tomados de distintas fuentes y llena una plantilla que define el documento oficial. Se sigue el formato de la guía de auditoría para su llenado. La plantilla es un documento en Word.	Revisiones válidas Subtítulos Descripciones

Proceso	2	
Nombre	Iniciar Auditoría	
Entradas	Descripción del proceso	Salidas
Revisiones a realizar Datos de orden de auditoría y acta de inicio	El auditor inicia la auditoría realizando los documentos de orden de auditoría y acta de inicio. Las auditorías se definen como nuevas o de seguimiento	Orden de auditoría Acta de inicio

Proceso	2.1	
Nombre	Elaborar Orden de Auditoría	
Entradas	Descripción del proceso	Salidas
Datos de orden de auditoría dados por el auditor Revisión aprobada	El auditor elabora la orden de auditoría para notificar al área a auditar la próxima realización de la misma. Si la orden de auditoría es de seguimiento se consideran los datos del inventario de	Orden de auditoría Datos de Orden de Auditoría a almacenar

Responsable(s) de la auditoría	observaciones y el texto de la orden se obtiene de una plantilla diferente. Los datos son tomados de distintas fuentes y llenan una plantilla que define el documento oficial. Se sigue el formato de la guía de auditoría para su llenado. La plantilla es un documento en Word. Si los datos del documento ya pasaron por supervisión, se reciben las correcciones, se llevan cabo y se genera el documento definitivo además de guardarse en el almacén de auditorías.	Datos de participantes de orden
Atención de la orden		
Personal que recibe las copias		
Clase (Tipo) de auditoría		
Área a auditar		
Texto de la orden		
Datos de observaciones pendientes para orden de auditoría		
Orden de auditoría a corregir		

4.1.4.2.2 Flujo de datos



Flujos en diagrama de contexto

Nombre del flujo	Registro de auditorías a realizar	
Descripción	Registro de datos de auditorías a realizar	
	Fuente	Destino
	DIRECTOR DE SUPERVISIÓN Y AUDITORÍA	Elaborar PACA
Estructura de datos que viaja con el flujo		
Registro de datos de auditorías a realizar		

Nombre del flujo	Revisiones válidas	
Descripción	Revisiones válidas	
	Fuente	Destino
	Elaborar PACA	D1(PACA)
Estructura de datos que viaja con el flujo		
Revisiones válidas		

Nombre del flujo	Subtítulos	
Descripción	Subtítulos de las revisiones del PACA	
Fuente		Destino
Elaborar PACA		D2 (SUBTÍTULOS)
Estructura de datos que viaja con el flujo		
Subtítulos de las revisiones del PACA		

Nombre del flujo	Descripciones	
Descripción	Descripciones de las revisiones del PACA	
Fuente		Destino
Elaborar PACA		D3 (DESCRIPCIONES)
Estructura de datos que viaja con el flujo		
Descripciones de las revisiones del PACA		

Nombre del flujo	Revisiones a realizar	
Descripción	Número de revisión autorizado en el PACA	
Fuente		Destino
D1 (PACA)		ELABORAR ORDEN DE AUDITORÍA
Estructura de datos que viaja con el flujo		
Número de revisión autorizado en el PACA		

4.1.4.2.3 Almacenes de datos

--	--

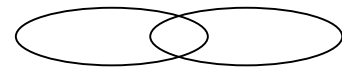
Nombre del almacén	D1(PACA)	
Descripción	Datos del programa anual de control de auditorías	
Flujos entrantes		Flujos salientes
Revisiones válidas		
Estructura de datos contenida en el almacén		
Revisiones		

Nombre del almacén	D2(SUBTÍTULOS)	
Descripción	Subtítulos de las revisiones	
Flujos entrantes		Flujos salientes
Subtítulos		
Estructura de datos contenida en el almacén		
Subtítulos contenidos en las revisiones		

Nombre del almacén	D3(DESCRIPCIONES)	
Descripción	Descripciones de la revisión	
Flujos entrantes		Flujos salientes
Descripciones		
Estructura de datos contenida en el almacén		
Descripciones contenidas en los subtítulos de las revisiones		

Nombre del Almacén	D4 (INVENTARIO DE OBSERVACIONES)	
Descripción	Almacén de observaciones para darles seguimiento	
Flujos entrantes		Flujos salientes
Datos de cédulas de observaciones a almacenar		Datos de Observaciones pendientes para ordenes de auditoría de seguimiento
Datos de cédulas de seguimientos a almacenar		Observaciones pendientes
Estructura de datos contenida en el almacén		
Observaciones nuevas y con seguimiento		

4.1.4.2.4 Estructuras de datos



La última etapa de la construcción del diccionario de datos es de hecho la que lo determina. A continuación se muestran las estructuras de datos con sus datos elementales resultado de la eliminación de redundancia entre ellas y que nos dan una descripción general de los datos que se emplearán en “SAPA” (basándose en ellos se construirá el modelo de E/R):

Nombre de la estructura		Revisiones		
Nombre	Tipo de dato	Longitud	Nulos	Descripción
Número progresivo de revisión	Alfanumérico	7	No	Número de progresivo de revisión con formato ####/# donde los dos primeros dígitos son el número de revisión consecutivo y los siguientes cuatro son el año en el que se lleva a cabo la auditoría
Fecha Paca	Fecha	10	No	Año de ejecución del PACA
Objetivo	Alfanumérico	1000	Si	Objetivo de cada una de las revisiones
Título	Alfanumérico	1000	Si	Título de cada una de las revisiones
Clave del Programa	Alfanumérico	5	Si	Clave del programa de cada una de las revisiones
Fecha de elaboración del paca	Fecha	10	No	Fecha de elaboración del PACA que normalmente es en noviembre del año anterior al año de ejecución del PACA

Nombre de la estructura		Subtítulos contenidos en las revisiones		
Nombre	Tipo de dato	Longitud	Nulos	Descripción
Subtítulos	Alfanumérico	500	Si	Varios Subtítulos contenidos en las revisiones

Nombre de la estructura		Descripciones contenidas en los subtítulos de las revisiones		
Nombre	Tipo de dato	Longitud	Nulos	Descripción
Descripciones	Alfanumérico	2000	Si	Varias descripciones contenidas en las revisiones

Nombre de la estructura		Observaciones nuevas y con seguimiento		
Nombre	Tipo de dato	Longitud	Nulos	Descripción
Número de observación	Numérico	2	No	Número consecutivo de la observación
Clasificación de la observación	Alfabético	18	Si	Clasificación de la observación
Título de la observación	Alfanumérico	200	Si	Título de la observación
Irregularidad	Alfanumérico	1000	Si	Irregularidad que genera la observación
Causa	Alfanumérico	1000	Si	Causa que genera la observación
Efecto	Alfanumérico	1000	Si	Efecto de la observación
Fundamento Legal	Alfanumérico	1000	Si	Fundamento legal de la observación
Recomendación Correctiva	Alfanumérico	1000	Si	Recomendación correctiva para la observación
Recomendación Preventiva	Alfanumérico	1000	Si	Recomendación Preventiva para la observación
Fecha de discusión	Fecha	10	Si	Fecha de discusión de la observación con formato dd/mm/aaaa
Fecha de implementación	Fecha	10	Si	Fecha de Implementación de la observación con formato dd/mm/aaaa
Monto fiscalizable	Alfanumérico	20	Si	Monto fiscalizable de la observación
Monto fiscalizador	Alfanumérico	20	Si	Monto fiscalizador de la observación
Monto irregularidad	Alfanumérico	20	Si	Monto irregularidad de la observación
Observación original	Alfanumérico	4000	Si	Campo que contiene los replanteamientos históricos de las observaciones.
Situación actual	Alfanumérico	1000	Si	Situación actual del seguimiento de la observación.
Recomendación preventiva (Replanteamiento)	Alfanumérico	1000	Si	Recomendaciones preventivas del seguimiento de la observación.
Recomendación correctiva (Replanteamiento)	Alfanumérico	1000	Si	Recomendaciones correctivas del seguimiento de la observación.

Fecha de Discusión de la Observación (Replanteamiento)	Fecha	10	Si	Fecha de Discusión del seguimiento de la observación. Fecha con formato dd/mm/aaaa.
Fecha de Implementación de la Observación (Replanteamiento)	Fecha	10	Si	Fecha de Implementación del seguimiento de la observación Fecha con formato dd/mm/aaaa
Status	Alfanumérico	1	No	Status del seguimiento de la observación
Avance	Numérico	3	Si	Avance del seguimiento

Nombre de la estructura		Datos del personal		
Nombre	Tipo de dato	Longitud	Nulos	Descripción
Nombre	Alfanumérico	50	No	Nombre del personal con saludo (Lic., Ing., etc.)
Credencial	Numérico	4	No	Número entero menor 1000
Domicilio	Alfanumérico	200	Si	Texto del domicilio de la persona
Cargo	Alfanumérico	200	Si	Cargo de la persona
Iniciales	Alfanumérico	5	Si	Iniciales del nombre de la persona
RFC	Alfanumérico	18	Si	Registro federal de causante con homoclave
Origen	Alfanumérico	20	Si	Texto que define el lugar de origen de la persona
Estado Civil	Alfanumérico	20	Si	Estado civil
Edad	Numérico	2	Si	Número entero menor a 100
Área	Alfanumérico	80	Si	Área a la que pertenece
Ubicación	Alfanumérico	10	Si	Ubicación del área a la que pertenece

Nombre de la estructura		Datos de las clases de auditoría		
Nombre	Tipo de dato	Longitud	Nulos	Descripción
Descripción de la clase	Alfanumérico	200	Si	Descripción de la clase de auditoría

Nombre de la estructura		Datos de las áreas a auditar		
Nombre	Tipo de dato	Longitud	Nulos	Descripción
Nombre	Alfanumérico	80	No	Nombre del área a auditar
Ubicación	Alfanumérico	10	Si	Ubicación de las áreas a auditar (piso)

Nombre de la estructura		Datos completos de las auditorías		
Nombre	Tipo de dato	Longitud	Nulos	Descripción
No de Orden de auditoría	Alfanumérico	7	No	Número de auditoría con formato ####### donde los dos primeros dígitos son el número de auditoría consecutivo y los siguientes cuatro son el año en el que lleva a cabo la auditoría.
Fecha de elaboración de la Orden de Auditoría	Fecha	10	No	Fecha con formato dd/mm/aaaa.
Fecha inicio del período por revisar	Fecha	10	Si	Fecha con formato dd/mm/aaaa.
Fecha final del período por revisar	Fecha	10	Si	Fecha con formato dd/mm/aaaa.
Fecha de inicio de la duración de la revisión	Fecha	10	Si	Fecha con formato dd/mm/aaaa.
Fecha final de la duración de la revisión	Fecha	10	Si	Fecha con formato dd/mm/aaaa.
Objeto de la revisión	Alfanumérico	1000	Si	Texto
Número progresivo de revisión	Alfanumérico	7	No	Número progresivo de revisión con formato ####### donde los dos primeros dígitos son el número de revisión consecutivo y los siguientes cuatro son el año en el que lleva a cabo la auditoría
Descripción de la clase	Alfanumérico	200	Si	Clase de auditoría que se realizará
Nombre del área	Alfanumérico	200	Si	Área a auditar
Numero total de observaciones pendientes	Numérico	3	Si	Número total de observaciones pendientes a dar seguimiento
Ordenes de auditoría de seguimiento	Alfanumérico	100	Si	Últimas órdenes de auditoría que dieron seguimiento a esas observaciones
Fecha de elaboración del acta de inicio	Fecha	10	Si	Fecha en la que se elabora el Acta de Inicio
Hora inicial de los hechos	Hora	8	Si	Hora inicial de los hechos mencionados en el Acta de Inicio
Hora final de los hechos	Hora	8	Si	Hora final de los hechos mencionados en el acta de inicio

Fecha de elaboración	Fecha	10	Si	Fecha de elaboración de la carta de planeación con formato dd/mm/aaaa
Trimestre	Alfanumérico	7	Si	Trimestre en el que se elabora la auditoría
Antecedentes del área	Alfanumérico	1000	Si	Antecedentes del área a auditar
Alcance	Alfanumérico	1000	Si	Alcance de la revisión
Problemática	Alfanumérico	1000	Si	Problemática
Estrategia	Alfanumérico	1000	Si	Estrategia para resolver la problemática
Número de Oficio	Alfanumérico	15	Si	Número de oficio del informe
Fecha del informe	Fecha	10	Si	Fecha del informe con formato dd/mm/aaaa
Informe Ejecutivo	Alfanumérico	1000	Si	Informe ejecutivo de la auditoría
Alcance	Alfanumérico	1000	Si	Alcance del informe
Resultado del trabajo	Alfanumérico	1000	Si	Resultado del trabajo en la auditoría
Conclusión y recomendación general	Alfanumérico	1000	Si	Conclusión y recomendación general de la auditoría
Tiempo real empleado	Numérico	2	Si	Tiempo en días/hombre empleado en la auditoría
Tiempo de supervisión	Numérico	2	Si	Tiempo en días/hombre empleado en la supervisión

Nombre de la estructura		Plantillas de los diferentes documentos a emplear en la auditoría		
Nombre	Tipo de dato	Longitud	Nulos	Descripción
Orden de auditoría	Alfanumérico	3000	No	Plantilla del documento
Acta de inicio	Alfanumérico	6000	No	Plantilla del documento
Carta planeación	Alfanumérico	3000	No	Plantilla del documento
Cronograma de actividades	Alfanumérico	3000	No	Plantilla del documento
Marco conceptual	Alfanumérico	1000	No	Plantilla del documento
Cédula de observaciones	Alfanumérico	2000	No	Plantilla del documento
Cédula de seguimientos	Alfanumérico	2000	No	Plantilla del documento
Informe de auditoría	Alfanumérico	3000	No	Plantilla del documento
Cédula única	Alfanumérico	1000	No	Plantilla del documento
Evaluación de auditoría	Alfanumérico	4000	No	Plantilla del documento

Nombre de la estructura		Almacén de datos de Cédulas de Marco Conceptual		
Nombre	Tipo de dato	Longitud	Nulos	Descripción
Número de Cédula	Alfanumérico	3	No	Número de la cédula realizada para el marco conceptual. Este número puede contener letras y números (4E, 7H, etc.)
Fecha del Marco	Fecha	10	Si	Fecha de realización de la cédula del marco conceptual con formato dd/mm/aaaa
Objetivo	Alfanumérico	500	Si	Objetivo de la cédula del marco conceptual
Universo	Alfanumérico	500	Si	Universo de la cédula del marco conceptual
Alcance	Alfanumérico	500	Si	Alcance de la cédula del marco conceptual
Fuente	Alfanumérico	500	Si	Fuente de la cédula del marco conceptual
Procedimiento	Alfanumérico	500	Si	Procedimiento de la cédula del marco conceptual
Conclusión	Alfanumérico	500	Si	Conclusión de la cédula del marco conceptual

Nombre de la estructura		Datos de los participantes en la orden de auditoría		
Nombre	Tipo de dato	Longitud	Nulos	Descripción
Nombre	Alfanumérico	50	Si	Nombre de los participantes en una auditoría sin saluto
Cargo	Alfanumérico	200	Si	Cargo de los participantes en una auditoría.
Ubicación	Alfanumérico	10	Si	Ubicación de los participantes en una auditoría.
Credencial	Alfanumérico	20	Si	Credencial de los participantes en una auditoría.
Edad	Alfanumérico	3	Si	Edad de los participantes en una auditoría.
Estado civil	Alfanumérico	20	Si	Estado Civil de los participantes en una auditoría.
Origen	Alfanumérico	20	Si	Origen de los participantes en una auditoría.
Domicilio	Alfanumérico	200	Si	Domicilio de los participantes en una auditoría.
RFC	Alfanumérico	18	Si	RFC de los participantes en una auditoría.

4.2 Diseño

4.2.1 Diseño de Carta Estructurada

Para la creación de la carta estructurada es necesario considerar dos aspectos importantes: el diseño de sistemas descendente y el desarrollo modular del software.

4.2.1.1 Diseño descendente de sistemas

Este enfoque de diseño de sistemas para una empresa consiste en visualizar primero un sistema general que considere los objetivos globales de la organización, y después desglosarlo en subsistemas.

El diseño descendente tiene tres ventajas principales:

- Se evita el tratar de diseñar un gran sistema en un sólo paso (esto porque hay que explotar cada subsistema).
- Se puede contar con grupos de analistas enfocados a cada subsistema (lo que incrementará la calidad del producto).
- Previene que el analista de sistemas se adentre demasiado en los detalles irrelevantes de los subsistemas al tener claro los objetivos centrales del gran sistema.

4.2.1.2 Desarrollo modular del software

Significa descomponer el desarrollo del software en unidades lógicas y manejables.

Esta manera de organizar el desarrollo de la programación posee cuatro ventajas:

- Los módulos son más fáciles de escribir y de revisar.
- La detección de errores es más sencilla.
- El mantenimiento de los sistemas modulares es más sencillo pues normalmente sólo es necesario modificar algunos módulos para lograr un cambio significativo.
- Son programas mucho más fáciles de comprender.

4.2.1.3 Diseño de la carta estructurada

El instrumento recomendado para el diseño de un sistema modular descendente es “La Carta Estructurada”. La carta estructurada muestra relaciones entre los módulos de un programa (representados por rectángulos) y que se conectan por medio de flechas. Los módulos se numeran y, si son submódulos, se emplea el mismo número del módulo padre más un punto y el número de submódulo. Al ser una técnica que modela diseño descendente, las flechas se dibujan con dirección de arriba abajo. Los símbolos que se emplean en el diagrama estructural son:

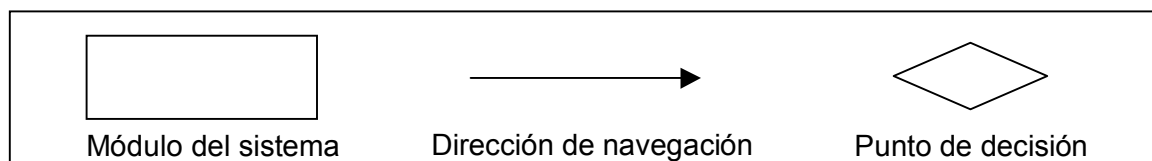


Figura 4.3 Elementos de la carta estructurada

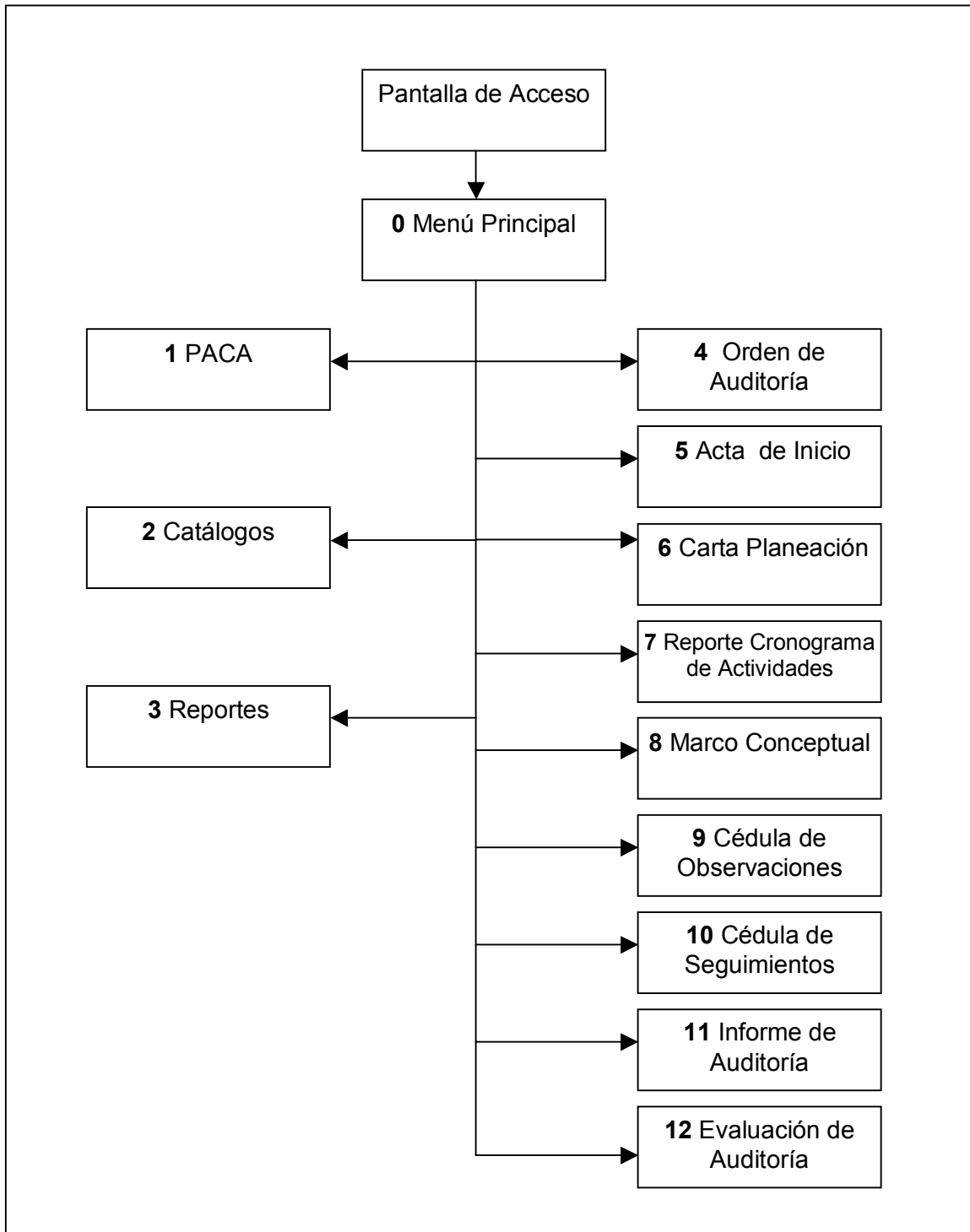


Figura 4.4 Carta estructurada

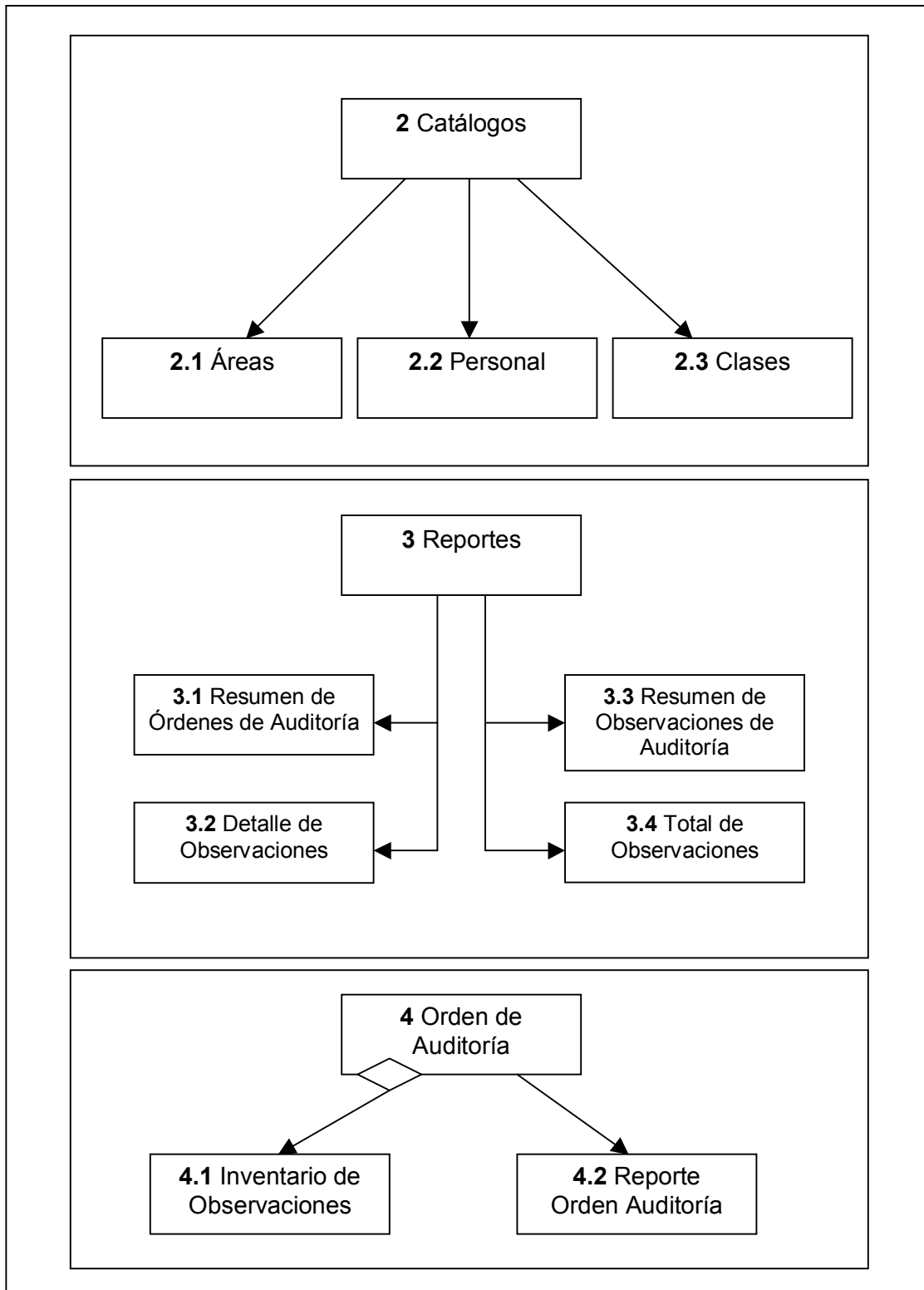


Figura 4.5 Carta estructurada (continuación)

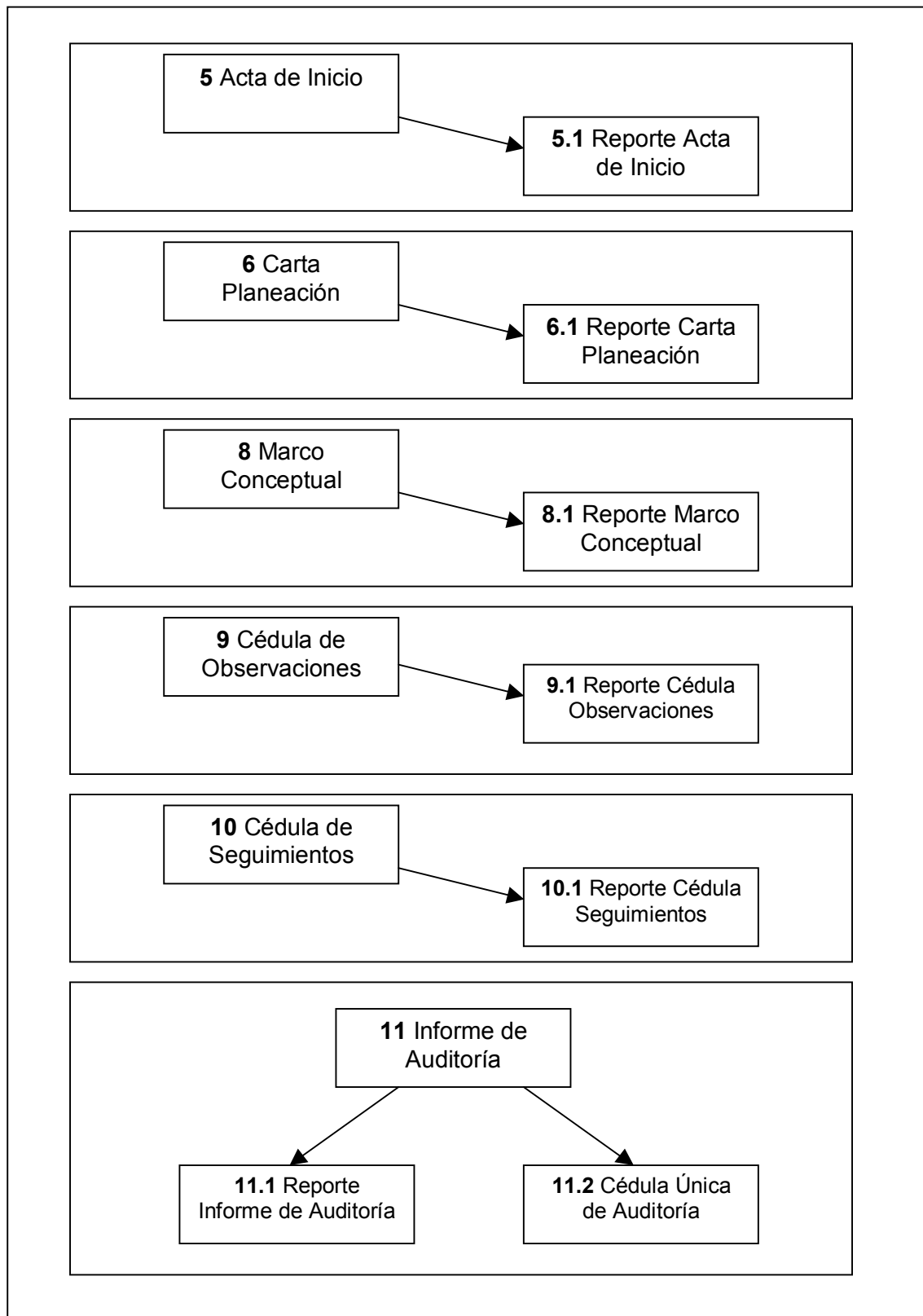


Figura 4.6 Carta estructurada (continuación)

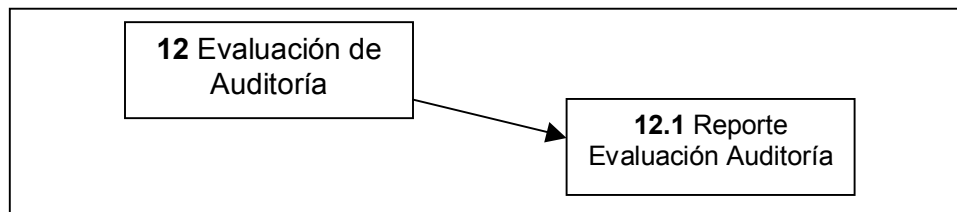


Figura 4.7 Carta estructurada (continuación)

4.2.2 Diseño de Procedimientos de Captura

Esta etapa se refiere a la captura de datos con efectividad, asegurando su calidad. En la etapa de captura de datos se pueden obtener grandes beneficios de productividad al reducir los tiempos en los que se introduce la información al sistema (dependiendo de la tecnología con la que se capturen), minimizar la captura de información redundante y evitar los costos de errores de corrección de datos almacenados incorrectamente.

Cuando se decide capturar se debe evitar la captura de datos redundantes, es decir, si en alguna parte del sistema se capturaron ciertos datos que son constantes durante el empleo de la aplicación (caso concreto en “SAPA”: catálogos de área, clases y personal), debe evitarse la captura repetitiva en diferentes módulos.

Otro aspecto a considerar en la captura de datos es el método por el cuál se va a realizar la captura. Para lograr determinar el método de captura adecuado será necesario considerar los factores de velocidad, precisión y la capacitación del operador, entre otros. Actualmente existen varios métodos:

- Teclado-almacenamiento.- Este método consiste en la introducción de datos por medio de un teclado que envía los datos a un medio de almacenamiento (disco, disco flexible). Actualmente este es el más popular de los métodos de captura de datos en las aplicaciones visuales cliente-servidor. “SAPA” hace uso principalmente de este método.
- Reconocimiento óptico de caracteres.- Este método permite acelerar la captura de documentos vía un OCR (reconocedor óptico de caracteres). La función del OCR es la de capturar documentos completos de manera óptica, es decir, en lugar de introducir carácter a carácter por medio del teclado, se reconoce y captura un documento completo en cuestión de segundos. El OCR se emplea de manera conjunta con un escáner que realiza la lectura óptica mientras que el OCR realiza la conversión de información óptica a caracteres.
- Archivos provenientes de otros sistemas.- Este método consiste en vaciar la información contenida en archivos que generan otros sistemas en la base de datos o archivos de información del sistema. Este método de captura optimiza la interacción del sistema a desarrollar con los otros sistemas ya existentes. “SAPA” considerará más adelante esta opción para obtener la información del PACA.
- Formas sensibles a marcas.- Este método combina el uso de formas con un reconocedor óptico de marcas. El funcionamiento de este método se observa en las formas de respuesta de cuestionarios de opción múltiple que son resueltos tachando una de las opciones con

lápiz. El reconocedor óptico detecta el plomo de las marcas de lápiz y envía la información a la computadora para ser procesada y/o almacenada.

- Códigos de barras.- Los códigos de barras tienen la función práctica de evitar al usuario la captura de datos para la búsqueda de información ya contenida en una base de datos de un inventario. Un ejemplo muy común de su uso se da en los productos de una tienda o en la identificación de pacientes de un hospital. Estos consisten de una serie de bandas angostas y anchas en una etiqueta, las cuales codifican números y/o letras.

4.2.3 Diseño de la Base de Datos

El modelo elegido en este sistema para la representación y estructuración lógica de los datos es el modelo relacional. Un elemento fundamental en el modelo relacional es el Diagrama de entidad relación (DER) que permite representar los objetos de datos de un sistema. El DER contiene los elementos básicos del modelado de datos: objetos de datos (entidades), atributos y relaciones.

Una entidad es un objeto o evento acerca del cual se recolectan datos. Puede ser una persona, un lugar o un objeto. Por ejemplo, un vendedor, una ciudad o un producto. También puede ser un evento o unidad de tiempo, tal como una venta, un mes o un año.

Las relaciones son asociaciones entre entidades. Existen tres tipos de relaciones:

- Relación de uno a uno (1:1). Un ejemplo sería que un empleado sólo pertenece a un departamento.
- Relación de uno a muchos (1:M). Un ejemplo de este caso es que un médico tiene varios pacientes.
- Relación de muchos a muchos (M:M). Un ejemplo de este caso es cuando un estudiante toma muchos cursos y a su vez un curso es impartido a varios estudiantes.

Los atributos son las características de las entidades. Una entidad puede poseer varios atributos. Por ejemplo una entidad estudiante puede poseer los atributos nombre, apellido, edad, etc. Los elementos y datos, mencionados en el diccionario de datos, deben ser considerados como atributos.

Es importante recordar que una entidad es cualquier composición de información o concepto que deba comprender el software, que los atributos definen las propiedades de las entidades y que las relaciones “interconectan” a las entidades. La notación del DER consiste en representar a las entidades con un rectángulo etiquetado conteniendo sus atributos llave en la parte superior y el resto en la parte inferior. Las relaciones se indican con líneas etiquetadas que unen a las entidades.

Es muy importante considerar que para la elaboración de un DER será necesario contar con un Diagrama de Flujo de Datos (DFD) y un Diccionario de Datos depurado.

4.2.3.1 Normalización

La “normalización” es el proceso por medio del cual se obtiene una base de datos con entidades relacionales, sin datos redundantes, consistente y que se puede modificar correctamente. Las ventajas de la normalización sobre las estructuras de datos son las siguientes:

- Estabilidad.
- Menor tamaño.
- Facilidad de mantenimiento.

Las reglas que deben de cumplir las entidades relacionales se conocen como formas normales. Mientras mayor es la forma normal, mejor es el diseño de la base de datos. En la práctica, un diseño que se lleva como máximo hasta la tercera forma normal se considera suficientemente bueno. Las primeras tres formas normales se describen a continuación.

4.2.3.1.1 Primera forma normal (FN1)

Consiste en la eliminación de grupos repetidos y la identificación de la llave que define el criterio primario. Con este fin, la relación inicial resulta ser dividida en dos o más relaciones. Como ejemplo se muestra la tabla “PACA”. En este caso fue necesario aplicar la primera forma normal.

No_Prog_Revisión	Fecha_Paca	Objetivo	Título	Cve_Programa	Fecha_P_Elab	Subtítulo	Descripción
01/2000	01/01/2000	Texto	Presupuesto-gasto corriente	350	03/11/99	Planeación	Texto1
01/2000	01/01/2000	Texto	Presupuesto-gasto corriente	350	03/11/99	Planeación	Texto2
01/2000	01/01/2000	Texto	Presupuesto-gasto corriente	350	03/11/99	Planeación	Texto3
01/2000	01/01/2000	Texto	Presupuesto-gasto corriente	350	03/11/99	Organización	Texto4
01/2000	01/01/2000	Texto	Presupuesto-gasto corriente	350	03/11/99	Organización	Texto5
06/1999	01/01/1999	Texto	Integrales	100	01/01/1999	Pago de honorarios	Texto 6
06/1999	01/01/1999	Texto	Integrales	100	01/01/1999	Pago de honorarios	Texto 7

Tabla 4.3 Tabla PACA

Esta entidad (tabla) almacena revisiones pertenecientes al Programa Anual de Control y Auditoría (PACA) y posee grupos repetidos: Subtítulo y Descripción. En primer lugar, existen varias descripciones por subtítulo, lo que llevó a crear una entidad descripción. También cada revisión posee varios subtítulos. De las conclusiones anteriores se obtuvieron las siguientes tablas:

No_prog_revisión	Fecha_paca	Objetivo	Título	Cve_programa	Fecha_p_elab
01/2000	01/01/2000	Texto	Presupuesto-gasto corriente	350	03/11/99
06/1999	01/01/1999	Texto	Integrales	100	01/01/1999

Tabla 4.4 Tabla “PACA” (Primera forma normal)

No_Prog_Revisión	Fecha_Paca	Num_Subtitulo	Subtítulo
01/2000	01/01/2000	1	Planeación
01/2000	01/01/2000	2	Organización
06/1999	01/01/1999	1	Pago de honorarios

Tabla 4.5 Tabla "SUBTITULO_PACA" (Primera forma normal)

No_Prog_Revisión	Fecha_Paca	Num_Subtitulo	No_Consec_Desc	Descripción
01/2000	01/01/2000	1	1	Texto1
01/2000	01/01/2000	1	2	Texto2
01/2000	01/01/2000	1	3	Texto3
01/2000	01/01/2000	2	1	Texto4
01/2000	01/01/2000	2	2	Texto5
06/1999	01/01/1999	1	1	Texto 6
06/1999	01/01/1999	1	2	Texto 7

Tabla 4.6 Tabla "DESCRIPCION_PACA" (Primera forma normal)

Al observar la entidad descompuesta en otras tres, se concluyó que en la entidad "PACA" sólo se almacenarán los datos generales de cada revisión, en la entidad "SUBTITULO_PACA" se almacenarán los subtítulos o Rubros que posee una revisión y que pueden ser varios por revisión. A su vez cada subtítulo posee varias descripciones que son contenidas en la entidad "DESCRIPCION_PACA".

4.2.3.1.2 Segunda forma normal (FN2)

En este paso se asegura que todos los atributos no-llave, sean completamente dependientes de la llave del criterio o atributo primario. Se dice entonces que un modelo de entidad relación está en la segunda forma normal si cumple con los criterios de la primera forma normal y los de la segunda al mismo tiempo. Por ejemplo en la entidad "SEGURIDAD" que se muestra a continuación, los atributos "Id_Personal", "Password" y "Tipo_Permission" son dependientes del atributo primario Login. Lo opuesto no es verdadero, es decir el atributo Login no depende de los demás atributos. Por lo anterior la Tabla "SEGURIDAD" cumple con la segunda forma normal.

Login	Id_Personal	Password	Tipo_Permission
ALOPEZ	5	080566	ADMIN
MCORTEZ	2	090972	USR

Tabla 4.7 Tabla "SEGURIDAD"(Segunda forma normal)

4.2.3.1.3 Tercera forma normal (FN3)

En este paso se eliminan todas las dependencias transitorias. Una dependencia transitoria es aquella en la que uno o varios atributos no llave son dependientes de otro atributo no llave. En “SAPA” no fue necesario el empleo de la tercera forma normal por lo que el ejemplo que a continuación se muestra no tiene que ver con el sistema.

CLAVE_TRAMITE	STATUS_TRAMITE	CLAVE_ALUMNO	NOMBRE_ALUMNO	FECHA_TRAMITE
5678	Concluido	34	José Llama	01/05/2001
7834	Concluido	23	Berenice Luna	04/05/2001
2389	Pendiente	34	José Llama	06/05/2001

Tabla 4.8 Tabla TRÁMITE _ ALUMNO

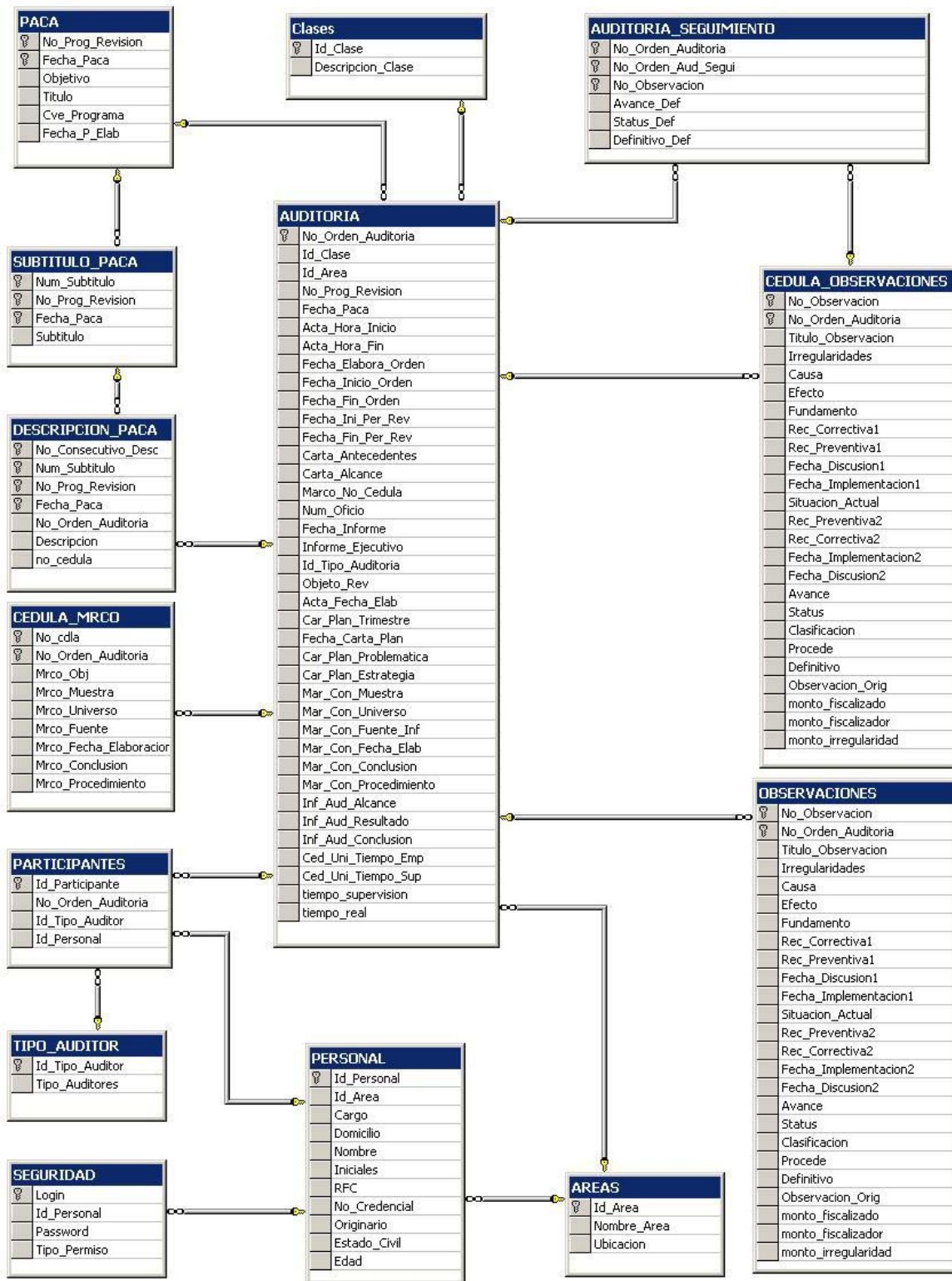
Al aplicar la tercera forma normal a esta tabla se puede ver que el atributo “Nombre_Alumno” depende del atributo no primario “Clave_Alumno”, por lo tanto se pueden crear las entidades Tramite y Alumno que cumplirán con la tercera forma normal, en donde no existirá una dependencia transitoria.

CLAVE_TRAMITE	STATUS_TRAMITE	CLAVE_ALUMNO	FECHA_TRAMITE
5678	Concluido	34	01/05/2001
7834	Concluido	23	04/05/2001
2389	Pendiente	34	06/05/2001

CLAVE_ALUMNO	NOMBRE_ALUMNO
34	José Llama
23	Verenice Sol

Tabla 4.9 Tablas TRÁMITE y ALUMNO (Tercera forma normal)

A continuación se muestra el DER normalizado (FN3) de “SAPA”:



4.2.4 Diseño de Entradas e Interfaz (Pantallas)

La calidad de la salida de un sistema está determinada por la calidad de la entrada de datos. Durante el diseño de las pantallas es importante tomar en cuenta lo anterior. El diseño de las pantallas debe satisfacer los siguientes objetivos: eficacia, precisión, facilidad de uso, consistencia, sencillez y atracción. Los objetivos anteriores se pueden satisfacer cuando se tiene conocimiento de lo que se requiere como entrada del sistema y cuando se comprende cómo reaccionará el usuario a los diferentes elementos de las pantallas.

Precisión significa que las pantallas de entrada satisfagan los propósitos específicos del sistema, mientras que eficacia significa que se realizarán estos propósitos. La facilidad de uso implica pantallas explícitas que no requerirán de tiempo adicional para descifrarse. La consistencia significa que los datos serán ordenados de manera parecida en las pantallas de la aplicación, y sencillez significa que el usuario mantendrá un mínimo de elementos dentro de las mismas que centren la atención del usuario y no la distraigan. Por último, la atracción significa que el usuario disfrute el diseño gráfico de las pantallas.

Para cumplir los objetivos anteriormente expuestos, existen cuatro lineamientos propuestos de desarrollo de las pantallas:

- *Mantener la pantalla sencilla.*- Este lineamiento se refiere a que la pantalla sólo debe de mostrar lo necesario para que una acción se lleve a cabo. Se propone la construcción de pantallas que contengan tres secciones: encabezado, cuerpo y pie o sección de comentarios e instrucciones. El encabezado debe contener información que indique al usuario en qué parte de la aplicación se encuentra (título de la aplicación y de la pantalla). Actualmente también se emplea esta sección para incluir menús de la aplicación. El cuerpo de la pantalla es la sección que se encuentra entre el encabezado y el pie. Esta sección contiene los campos en donde se consulta, actualiza o borra la información de la aplicación. La información se ordena normalmente de izquierda a derecha y de arriba hacia abajo. El pie o sección de comentarios es en donde se puede incluir un menú de instrucciones que recuerden al usuario las funciones básicas del sistema.
- *Mantener una presentación consistente.*- El primer concepto a considerar es que el diseño de una pantalla debe de apegarse lo más posible a los diseños de las formas a partir de las cuales se originan. A continuación será necesario agrupar la información que tenga una relación lógica (nombre, apellido paterno, etc.). Por último será importante que la información que aparece en la pantalla no varíe sino que sea consistente cada vez que sea empleada.
- *Facilitar los movimientos del usuario entre las pantallas.*- Este concepto implica poderse desplazar entre una pantalla y otra de manera sencilla. Los lenguajes visuales permiten crear pantallas en las cuales se pueden aplicar los siguientes métodos que simulan un movimiento físico a otra pantalla:

Desplazamiento.- Es cuando el tamaño del área de la pantalla no es suficiente para mostrar todos los campos que contiene la información que requiere una pantalla. En este caso la pantalla en realidad posee un tamaño superior al que puede mostrarse y los campos que

quedan ocultos se muestran a través de una barra de desplazamiento llamada scrollbar o por medio de los botones de dirección del teclado.

Ventanas de detalle.- Son ventanas que aparecen para mostrar mayor detalle de algún dato en la pantalla (el detalle de la información de un cliente por ejemplo).

Diálogos en pantalla.- También son ventanas, sólo que pueden aparecer como parte del proceso de captura, para mostrar mensajes de error, mensajes del sistema, carga de archivos, entre otras funciones.

- *Crear pantallas atractivas.*- Las pantallas deben mantener la atención del usuario. Una forma de lograrlo es evitando la sobrecarga de información, es decir, no debe haber un exceso de campos de información en una sola pantalla, mas bien debe haber un espacio razonable entre cada uno. Es mejor emplear pantallas múltiples que amontonar la información en una sola pantalla. El emplear pantallas que son fáciles de recordar atrae al usuario experto e inexperto. Se recomienda que en el caso de pantallas complejas se agrupen por medio de líneas (frames) las diferentes categorías de información. Existen diversas técnicas gráficas que se aplican para hacer más estético el diseño de una pantalla y varían dependiendo del lenguaje de programación elegido para el desarrollo, pero en general podemos hablar de las siguientes: video inverso y diferentes tipos de letras (*clipper*), imágenes y color (*Visual Basic y Delphi*), entre otras.

Para que el desarrollo de “SAPA” cumpliera con los objetivos de diseño de buenas pantallas de la aplicación, se definieron los siguientes estándares de diseño:

- Todas las formas deben tener márgenes verticales y horizontales de 150 twip, es decir, que ningún control puede colocarse sobre el área que definen los márgenes (también aplica para el TabCtl).
- En caso de que los botones tengan una orientación horizontal, los botones con **Caption Aceptar** y **Cancelar** deben ir alineados al lado derecho de la forma; los demás botones se alinearán de lado izquierdo. Si la orientación es vertical, los botones **Aceptar** y **Cancelar** deben ir alineados hacia la parte inferior de la forma, mientras que los demás deberán alinearse hacia la parte superior.
- La separación entre botones es la siguiente:
Verticalmente:
 - Del mismo grupo : 2 pixeles
 - De diferente grupo: 12 pixelesHorizontalmente:
 - Del mismo grupo : 4 pixeles
 - De diferente grupo: 12 pixeles
- La propiedad top de los controles Label debe valer 60 unidades mas que el top del control al que corresponden.
- Los Spread y los ListBoxes no deben ir dentro de un Frame, a menos que se agrupen con otros controles, además deben tener un Label que los identifique.
- Si una ventana maneja una gran cantidad de controles que pueden ser agrupados por conjuntos lógicos, entonces lo ideal será utilizar un TabCtl para optimizar el espacio en pantalla.
- Se deben utilizar separadores para indicar grupos dentro de un submenú.
- Toda acción que involucra abrir una ventana debe llevar la terminación ... en el Caption.

- Si una opción del menú representa un valor booleano, este debe indicarse con la presencia o no de una palomita (Menú checked).

Cumpliendo los estándares de diseño antes mencionados, se elaboró el diseño de las pantallas del sistema. A continuación se muestra como ejemplo el documento de diseño de la pantalla del módulo “PACA”:

Sistema	Sistema de Automatización del Proceso de Auditoría (SAPA)
Módulo	PACA (Programa Anual de Control y Auditoría)
Pantalla	FrmPACA
Proceso/Entrada	Elaborar PACA
Operación Básica	Registrar, modificar y consultar la información del PACA

Objetivo:

Permitir al usuario la captura, modificación y consulta del PACA de diferentes años.

Narrativa:

Esta pantalla es de tipo de registro y consulta de información:

1. El usuario podrá consultar en los edits y memos, la información de los PACAS capturados con anterioridad (Títulos, Rubros y Descripciones). Esta información está almacenada en las tablas PACA, SUBTITULO_PACA y DESCRIPCION_PACA.
2. El usuario podrá modificar el valor de esta información al presionar el botón cambios.
3. Al dar clic en el botón nuevo, el usuario podrá capturar la información correspondiente a nuevas revisiones (títulos, rubros y descripciones) para un PACA nuevo o uno existente.
4. Al dar clic en el botón Documento, el usuario podrá obtener tanto el PACA resumen como el PACA detallado en un documento de Excel.
5. Se podrán filtrar las revisiones por año PACA.

Datos en Pantalla

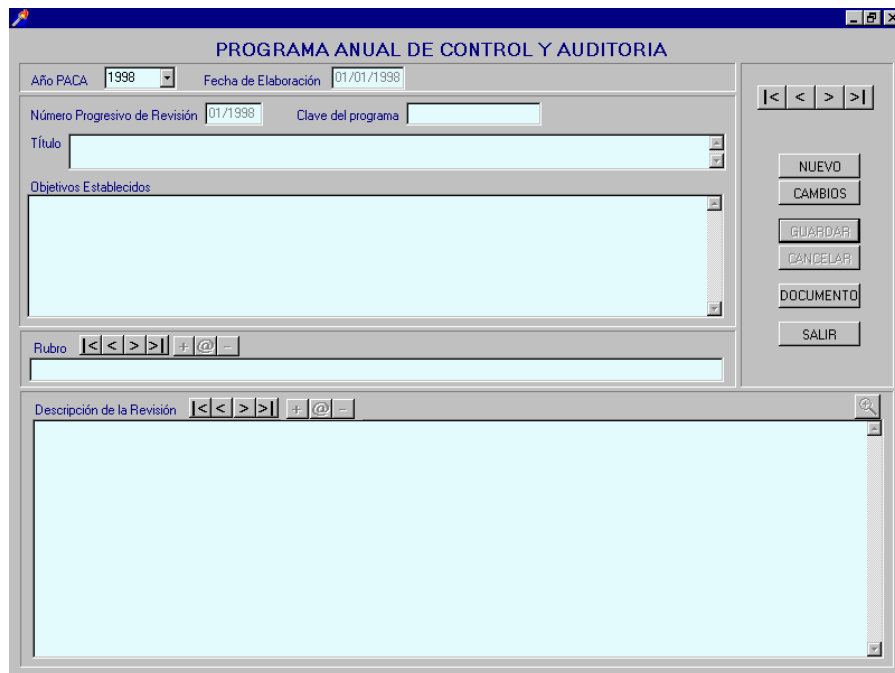
Dato en Pantalla	Tipo	Descripción	Campo en B. D.	Validación en Pantalla	Validación en Base de Datos
Año PACA	Combobox	Muestra la información de los años en los que se ejecutará el PACA	Fecha_paca	Estado inicial año del PACA Al hacer click Permite elegir el año del PACA a consultar	*
Fecha de Elaboración	MaskEdit	Muestra la información de la fecha en la que se elabora el PACA	Fecha_P_elab	Estado inicial lleno con la información del campo que se indica	*
Número progresivo de revisión	Edit	Número progresivo de revisión contenida en el PACA	No_Prog_Revision	Estado inicial lleno con la información del campo que se indica	*
Título	Memo	Titulo de las revisiones del PACA	Título	Estado inicial, lleno con el campo que se indica	*
Objetivos establecidos	Memo	Objetivos establecidos de la revisiones del PACA	Objetivo	Estado inicial, lleno con el campo que se indica	*
Rubro	Memo	Rubros de los títulos de las revisiones del PACA	Subtítulo	Al hacer click en el permite elegir una fecha basándose en el calendario	*
Descripción	Memo	Descripciones de los Rubros de los títulos de las revisiones del PACA	Descripción	Estado inicial, lleno con el campo que se indica	*

Sistema de Automatización del Proceso de Auditoría

<	Button	Permite consultar la primera revisión del PACA seleccionado	*	*	*
<	Button	Permite consultar la revisión anterior del PACA seleccionado	*	*	*
>	Button	Permite consultar la revisión siguiente del PACA seleccionado	*	*	*
>	Button	Permite consultar la última revisión del PACA seleccionado	*	*	*
NUEVO	Button	Activa la pantalla para captura una nueva revisión	*	Estado inicial habilitado. Se deshabilita en el momento en el que se captura revisión nueva o se hacen cambios.	*
CAMBIOS	Button	Activa la pantalla para realizar cambios en las revisiones existentes	*	Estado inicial habilitado. Se deshabilita en el momento en el que se captura revisión nueva o se hacen cambios.	*
GUARDAR	Button	Guarda los cambios o nuevos datos de las revisiones	*	Estado inicial no habilitado. Se habilita hasta que se presiona el botón nuevo o cambios.	*
CANCELAR	Button	Cancela los cambios o las nuevas revisiones	*	Estado inicial no habilitado. Se habilita hasta que se presiona el botón nuevo o cambios.	*
DOCUMENTO	Button	Genera el documento PACA Resumen o PACA Detallado	*	Estado inicial habilitado. Se deshabilita en el momento en el que se captura revisión nueva o se hacen cambios.	*
SALIR	Button	Permite salir del módulo PACA	*	Estado inicial habilitado. Se deshabilita en el momento en el que se captura revisión nueva o se hacen cambios.	*
<	Button	Permite consultar el primer rubro de la revisión seleccionada	*	*	*
<	Button	Permite consultar el rubro anterior de la revisión seleccionada	*	*	*
>	Button	Permite consultar el rubro siguiente de la Revisión seleccionada	*	*	*
>	Button	Permite consultar el último rubro de la revisión seleccionada	*	*	*
<	Button	Permite consultar el primer rubro de la revisión seleccionada	*	*	*
<	Button	Permite consultar el rubro anterior de la revisión seleccionada	*	*	*
>	Button	Permite consultar el rubro siguiente de la revisión seleccionada	*	*	*
>	Button	Permite consultar el último rubro de la revisión seleccionada	*	*	*
+	Button	Permite agregar Rubros a las revisiones	*	Estado inicial no habilitado. Se habilita hasta que se presiona el botón nuevo o cambios.	*
@	Button	Permite actualizar la información de los	*	Estado inicial no habilitado. Se habilita hasta que se	*

		rubros existentes		presiona el botón nuevo o cambios.	
-	Button	Permite borrar rubros de las revisiones	*	Estado inicial no habilitado. Se habilita hasta que se presiona el botón nuevo o cambios.	*
<	Button	Permite consultar la primera descripción del rubro seleccionado	*	*	*
<	Button	Permite consultar la descripción anterior del rubro seleccionado	*	*	*
>	Button	Permite consultar la descripción siguiente del Rubro seleccionado	*	*	*
>	Button	Permite consultar la última descripción del Rubro seleccionado	*	*	*
+	Button	Permite agregar descripciones a los Rubros	*	Estado inicial no habilitado. Se habilita hasta que se presiona el botón nuevo o cambios.	*
@	Button	Permite actualizar la información de las descripciones existentes	*	Estado inicial no habilitado. Se habilita hasta que se presiona el botón nuevo o cambios.	*
-	Button	Permite borrar descripciones de los rubros	*	Estado inicial no habilitado. Se habilita hasta que se presiona el botón nuevo o cambios.	*
Zoom	BitBtn	Permite maximizar el área de consulta de las descripciones.	*	Estado inicial no habilitado. Se habilita hasta que se presiona el botón nuevo o cambios	*

Pantalla



DESCRIPCIÓN DE TABLAS DE LA BD					
Nombre	Descripción	Tipo de acceso			
		Select	Insert	Delete	Update
PACA	Datos de las revisiones de los PACA	X	X		X
Subtitulo_PACA	Datos de los rubros de las revisiones de los PACA	X	X	X	X
Descripcion_PACA	Datos de las descripciones de los rubros	X	X	X	X

Rutinas reutilizables	
Número	Descripción

Algoritmos

Estado inicial

1. Llenar combo de año PACA con una lista de los valores de los diferentes año PACA contenidos en el campo Fecha_PACA de la tabla PACA.
2. Obtener Fecha_P_Elab, No_Prog_Revision, Cve_programa, Titulo, Objetivo, Subtitulo y Descripción de la tablas Paca, Subtitulo_PACA y Descripcion_PACA para inicializar los campos.
3. Inicializar los controles con los datos que obtuvo de BD, de la siguiente manera:
 - 3.1. Seleccionar el elemento del combo Año PACA que coincida con el valor del campo Fecha_Paca.
 - 3.2. Fecha de elaboración = Fecha_P_Elab de la tabla PACA.
 - 3.3. Número progresivo de revisión = No_Prog_Revision de la tabla PACA.
 - 3.4. Clave del programa = Cve_programa de la tabla PACA.
 - 3.5. Título = Título de la tabla PACA.
 - 3.6. Objetivos establecidos = Objetivo de la tabla PACA.
 - 3.7. Rubro = Subtítulo de la tabla SUBTITULO_PACA.
 - 3.8. Descripción de la Revisión = Descripcion de la tabla DESCRIPCION_PACA
4. Se deben deshabilitar los controles:
 - 4.1. MaskEdit Edit Fecha de elaboración.
 - 4.2. Edit Número progresivo de revisión.
 - 4.3. Edit Clave del programa.
 - 4.4. Memo Título.
 - 4.5. Memo Objetivos establecidos.
 - 4.6. Memo Rubro.
 - 4.7. Memo Descripción de la Revisión.

Combo Año PACA

1. Se selecciona un año de la lista que presenta el combo año PACA.
2. Se realiza una consulta de Fecha_P_Elab, No_Prog_Revision, Cve_programa, Titulo, Objetivo, Subtítulo y Descripción de la tablas Paca, Subtitulo_PACA y

Descripcion_PACA empleando como parámetro de filtrado Fecha_PACA obtenido del mismo control combo Año PACA.

3. Inicializar los controles con los datos que obtuvo de BD, de la siguiente manera:
 - 3.1. Seleccionar el elemento del combo Año PACA que coincida con el valor del campo Fecha_Paca.
 - 3.2. Fecha de elaboración = Fecha_P_Elab de la tabla PACA.
 - 3.3. Número progresivo de revisión = No_Prog_Revision de la tabla PACA.
 - 3.4. Clave del programa = Cve_programa de la tabla PACA.
 - 3.5. Título = Título de la tabla PACA.
 - 3.6. Objetivos establecidos = Objetivo de la tabla PACA.
 - 3.7. Rubro = Subtítulo de la tabla SUBTITULO_PACA.
 - 3.8. Descripción de la Revisión = Descripcion de la tabla DESCRIPCION_PACA

Botón Nuevo

1. Al presionar el botón nuevo deberán limpiarse y habilitarse los siguientes controles:
 - 1.1 MaskEdit Edit Fecha de elaboración.
 - 1.2 Edit Número progresivo de revisión.
 - 1.3 Edit Clave del programa.
 - 1.4 Memo Título.
 - 1.5 Memo Objetivos establecidos.
 - 1.6 Memo Rubro.
 - 1.7 Memo Descripción de la Revisión.
2. También deberán habilitarse los siguientes controles
 - 2.1. Button (+) de rubros
 - 2.2. Button (-) de rubros
 - 2.3. Button (@) de rubros
 - 2.4. Button (+) de descripciones
 - 2.5. Button (-) de descripciones
 - 2.6. Button (@) de descripciones
 - 2.7. Button Guardar
 - 2.8. Button Cancelar
 - 2.9. BitBtn Zoom
3. Por último deberán deshabilitarse los botones
 - 3.1. Button |< principal
 - 3.2. Button < principal
 - 3.3. Button > principal
 - 3.4. Button >| principal
 - 3.5. Button Documento
 - 3.6. Button Salir
4. Deberá colocarse el foco en el campo Año PACA.

Botón Cambios

1. Al presionar el botón nuevo habilitarse los siguientes controles:
 - 1.1 MaskEdit Edit Fecha de elaboración.
 - 1.2 Edit Clave del programa.
 - 1.3 Memo Título.

- 1.4 Memo Objetivos establecidos.
 - 1.5 Memo Rubro.
 - 1.6 Memo Descripción de la Revisión.
 - 1.7 Button (+) de rubros.
 - 1.8 Button (-) de rubros.
 - 1.9 Button (@) de rubros.
 - 1.10 Button (+) de descripciones.
 - 1.11 Button (-) de descripciones.
 - 1.12 Button (@) de descripciones.
 - 1.13 Button Guardar.
 - 1.14 Button Cancelar.
 - 1.15 BitBtn Zoom.
2. Por último deberán deshabilitarse los botones:
 - 2.1 Button |< principal
 - 2.2 Button < principal
 - 2.3 Button > principal
 - 2.4 Button >| principal
 - 2.5 Button Documento
 - 2.6 Button Salir
 3. Deberá colocarse el foco en el campo Clave del Programa.

Botón Guardar

1. Validar que lo siguiente:
 - 1.1. Verificar que la Fecha de elaboración no este vacía, con valores incompletos o no válidos (El formato de la fecha es de dd/mm/yyyy).
 - 1.2. Si no es así:
 - 1.2.1. Colocar el foco en el campo Fecha de elaboración.
 - 1.2.2. Mostrar mensaje de error M001.
 - 1.3. Verificar que los últimos cuatro dígitos del número progresivo de revisión coincidan con el año PACA.
 - 1.4. Si no es así:
 - 1.4.1. Colocar el foco en el campo número progresivo de revisión.
 - 1.4.2. Mostrar el mensaje de error M002.
 - 1.5. Verificar que el número progresivo de revisión no pertenezca a una revisión ya existente.
 - 1.6. Si no es así:
 - 1.6.1. Colocar el foco en el campo número progresivo de revisión.
 - 1.6.2. Mostrar mensaje de error M003.
2. Deben deshabilitarse los siguientes controles:
 - 2.1. MaskEdit Edit Fecha de elaboración.
 - 2.2. Edit Clave del programa.
 - 2.3. Memo Título.
 - 2.4. Memo Objetivos establecidos.
 - 2.5. Memo Rubro.
 - 2.6. Memo Descripción de la Revisión.
 - 2.7. Button (+) de rubros.

- 2.8. Button (-) de rubros.
- 2.9. Button (@) de rubros.
- 2.10. Button (+) de descripciones.
- 2.11. Button (-) de descripciones.
- 2.12. Button (@) de descripciones.
- 2.13. Button Guardar.
- 2.14. Button Cancelar.
- 2.15. BitBtn Zoom.
2. Deberán habilitarse los botones.
 - 2.1 Button |< principal
 - 2.2 Button < principal
 - 2.3 Button > principal
 - 2.4 Button >| principal
 - 2.5 Button Documento
 - 2.6 Button Salir
2. Obtener Fecha_P_Elab, No_Prog_Revision, Cve_programa, Título, Objetivo, Subtítulo y Descripción de la tablas Paca, Subtítulos y Descripciones para inicializar los campos.
3. Inicializar los controles con los datos que obtuvo de BD, de la siguiente manera:
 - 3.1. Seleccionar el elemento del combo Año PACA que coincida con el valor del campo Fecha_Paca.
 - 3.2. Fecha de elaboración = Fecha_P_Elab de la tabla PACA.
 - 3.3. Número progresivo de revisión = No_Prog_Revision de la tabla PACA.
 - 3.4. Clave del programa = Cve_programa de la tabla PACA.
 - 3.5. Título = Titulo de la tabla PACA.
 - 3.6. Objetivos establecidos = Objetivo de la tabla PACA.
 - 3.7. Rubro = Subtítulo de la tabla SUBTITULO_PACA.
4. Descripción de la Revisión = Descripcion de la tabla DESCRIPCION_PACA
5. Deberá colocarse el foco en el campo Año PACA

Botón Cancelar

1. Al presionar el botón cancelar deben deshabilitarse los siguientes controles:
 - 1.1 MaskEdit Edit Fecha de elaboración.
 - 1.2 Edit Clave del programa.
 - 1.3 Memo Título.
 - 1.4 Memo Objetivos establecidos.
 - 1.5 Memo Rubro.
 - 1.6 Memo Descripción de la Revisión.
 - 1.7 Button (+) de rubros.
 - 1.8 Button (-) de rubros.
 - 1.9 Button (@) de rubros.
 - 1.10 Button (+) de descripciones.
 - 1.11 Button (-) de descripciones.
 - 1.12 Button (@) de descripciones.
 - 1.13 Button Guardar.
 - 1.14 Button Cancelar.

- 1.15 BitBtn Zoom.
2. Deberán habilitarse los botones:
 - 2.1 Button |< principal.
 - 2.2 Button < principal.
 - 2.3 Button > principal.
 - 2.4 Button >| principal.
 - 2.5 Button Documento.
 - 2.6 Button Salir.
3. Obtener Fecha_P_Elab, No_Prog_Revision, Cve_programa, Titulo, Objetivo, Subtítulo y Descripción de la tablas PACA, Subtitulos_PACA y Descripción_PACA para inicializar los campos.
4. Inicializar los controles con los datos que obtuvo de BD, de la siguiente manera:
 - 4.1. Seleccionar el elemento del combo Año PACA que coincida con el valor del campo Fecha_Paca.
 - 4.2. Fecha de elaboración = Fecha_P_Elab de la tabla PACA.
 - 4.3. Número progresivo de revisión = No_Prog_Revision de la tabla PACA.
 - 4.4. Clave del programa = Cve_programa de la tabla PACA.
 - 4.5. Título = Título de la tabla PACA.
 - 4.6. Objetivos establecidos = Objetivo de la tabla PACA.
 - 4.7. Rubro = Subtítulo de la tabla SUBTITULO_PACA.
 - 4.8. Descripción de la Revisión = Descripción de la tabla DESCRIPCION_PACA.
5. Deberá colocarse el foco en el campo Año PACA.

Botón Documento

1. Muestra un diálogo en el que se puede elegir lanzar el reporte de PACA Resumen o PACA Detallado o no mandar ningún reporte.
2. Si se elige Lanzar reporte PACA Detallado, ejecuta la consulta de los campos No_Prog_Revision, Objetivo, Titulo, Cve_Programa, Num_Subtitulo, Subtitulo, No_Consecutivo_Desc, Descripción, Fecha_Paca y Fecha_P_Elab de las tablas PACA, Subtitulo_PACA y Descripción_PACA y vacía los resultados en un formato del PACA resumen diseñado en Excel.
3. Si se elige Lanzar reporte PACA Resumen, ejecuta la consulta de los campos No_Prog_Revision, Título, Cve_Programa, Fecha_Paca y Fecha_P_Elab de la tabla PACA y vacía los resultados en un formato del PACA resumen diseñado en Excel.

Botones |<, <, >, >| de sección principal.

1. Botones de navegación (primero, anterior, siguiente y último) entre los registros de la consulta Fecha_P_Elab, No_Prog_Revision, Cve_programa, Titulo, Objetivo, Subtítulo y Descripción de la tablas PACA, Subtitulo_PACA, y Descripción_PACA.

Botones |<, <, >, >| de sección rubro.

1. Botones de navegación (primero, anterior, siguiente y último) entre los registros de la consulta Subtítulo y Descripción de la tablas Subtitulo_PACA, y Descripción_PACA correspondientes a la revisión actual.

Botones |<, <, >, >| de sección descripción de la revisión.

1. Botones de navegación (primero, anterior, siguiente y último) entre los registros de la consulta Descripción de la tablas Descripción_PACA para inicializar los campos correspondientes al rubro seleccionado.

Botón + de la sección rubro.

1. Verificar que esté capturado Año PACA.
2. Si no es así:
 - 2.1. Mandar mensaje de error M004
 - 2.2. Colocar el cursor en el campo Año PACA.
3. Verificar que esté capturado No. Progresivo de Revisión.
4. Si no es así:
 - 4.1. Mandar mensaje de error M004.
 - 4.2. Colocar el cursor en el campo No. Progresivo de Revisión.
5. Verificar que los últimos cuatro dígitos del número progresivo de revisión coincidan con el año PACA.
 - 5.1. Si no es así:
 - 5.1.1. Colocar el foco en el campo número progresivo de revisión.
 - 5.1.2. Mostrar el mensaje de error M002.
6. Verificar que el número progresivo de revisión no pertenezca a una revisión ya existente.
 - 6.1. Si no es así:
 - 6.1.1. Colocar el foco en el campo número progresivo de revisión.
 - 6.1.2. Mostrar mensaje de error M003.
7. Verificar que la Fecha de elaboración no esté vacía, con valores incompletos o no válidos (El formato de la fecha es de dd/mm/yyyy).
8. Si no es así:
 - 8.1. Colocar el foco en el campo Fecha de elaboración.
 - 8.2. Mostrar mensaje de error M001.
9. Inserta un nuevo registro en la tabla Subtitulo_PACA creando un nuevo Num_Subtitulo (consecutivo) e insertando el valor de Rubro en el campo Subtitulo. Emplea los parámetros No_Prog_Revisión y Fecha_PACA de la revisión actual.

Botón @ de la sección rubro.

1. Verificar que esté capturado un rubro.
2. Si no es así:
 - 2.1. Mandar mensaje de error M005.
 - 2.2. Colocar el cursor en el campo rubro.
3. Actualiza un registro en la tabla Subtitulo_PACA (valor de Rubro = Subtítulo). Emplea los parámetros Num_Subtitulo, No_Prog_Revisión y Fecha_PACA de la revisión actual.

Botón - de la sección rubro.

1. Mandar un mensaje de confirmación de borrado.
2. Si se elige si:
 - 2.1. Verificar que este capturado un rubro.
 - 2.2. Si no es así:
 - 2.2.1. Mandar mensaje de error M006.

- 2.2.2. Colocar el cursor en el campo rubro.
- 2.3. Borra un registro en la tabla Subtítulo_PACA empleando los parámetros Num_Subtitulo, No_Prog_Revisión y Fecha_PACA de la revisión actual.
3. Si se elige “no”:
 - 3.1. Colocar el foco en el campo rubro.

Botón + de la sección descripción de la revisión.

1. Verificar que esté capturado Año PACA.
2. Si no es así:
 - 2.1. Mandar mensaje de error M004.
 - 2.2. Colocar el cursor en el campo Año PACA.
3. Verificar que este capturado No. Progresivo de Revisión.
4. Si no es así:
 - 4.1. Mandar mensaje de error M004.
 - 4.2. Colocar el cursor en el campo No. Progresivo de Revisión.
5. Verificar que los últimos cuatro dígitos del número progresivo de revisión coincidan con el año PACA.
 - 5.1. Si no es así:
 - 5.1.1. Colocar el foco en el campo número progresivo de revisión.
 - 5.1.2. Mostrar el mensaje de error M002.
6. Verificar que el número progresivo de revisión no pertenezca a una revisión ya existente.
 - 6.1. Si no es así:
 - 6.1.1. Colocar el foco en el campo número progresivo de revisión.
 - 6.1.2. Mostrar mensaje de error M003.
7. Verificar que la Fecha de elaboración no esté vacía, con valores incompletos o no válidos (El formato de la fecha es de dd/mm/yyyy).
8. Si no es así:
 - 8.1. Colocar el foco en el campo Fecha de elaboración.
 - 8.2. Mostrar mensaje de error M001.
9. Verificar que este capturado un rubro.
10. Si no es así:
 - 10.1. Mandar mensaje de error M007.
 - 10.2. Colocar el cursor en el campo descripción de la revisión.
11. Inserta un nuevo registro en la tabla Descripcion_PACA creando un nuevo Num_Consecutivo_Desc (consecutivo) e insertando el valor de Descripción de la revisión en el campo Descripción. Emplea los parámetros Num_Subtitulo, No_Prog_Revisión y Fecha_PACA de la revisión actual.

Botón @ de la sección descripción de la revisión.

1. Verificar que esté capturado un rubro.
2. Si no es así:
 - 2.1. Mandar mensaje de error M005.
 - 2.2. Colocar el cursor en el campo Descripcion de la revisión.
3. Actualiza un registro en la tabla Descripcion_PACA (Descripción de la revisión = Descripción). Emplea los parámetros Num_Consecutivo_Desc, Num_Subtitulo, No_Prog_Revisión y Fecha_PACA.

Botón - de la sección descripción de la revisión.

1. Se manda un diálogo de confirmación de operación.
2. Si se elige “sí”:
 - 2.1. Verificar que este capturado un rubro.
 - 2.2. Si no es así:
 - 2.2.1. Mandar mensaje de error M006.
 - 2.2.2. Colocar el cursor en el campo descripción de la revisión.
 - 2.3. Borra un registro en la tabla Descripcion_PACA empleando los parámetros Num_Consecutivo_Desc, Num_Subtitulo, No_Prog_Revisión y Fecha_PACA.
3. Si se elige “no”:
 - 3.1. Se coloca el foco en el campo descripción de la revisión.

Mensajes	
Número	Descripción
M001	Teclea el número progresivo de revisión o la fecha de elaboración.
M002	No coincide el año PACA con el número progresivo de revisión.
M003	El Número progresivo de revisión ya está asignado a otra auditoría.
M004	Falta especificar año del PACA y Número progresivo de revisión.
M005	Error al modificar datos.
M006	Error al borrar datos.
M007	Falta Año del PACA, Número progresivo de revisión o Subtítulo.

4.2.5 Diseño de Salidas

La salida es la información que reciben los usuarios del sistema de información. Las salidas pueden tomar distintas formas: reportes impresos, pantallas en monitor, archivos, etc. Para los usuarios, las salidas son una herramienta para la realización de su trabajo. Por lo anterior muchas veces los usuarios evalúan el desempeño de un sistema a través de la calidad de sus salidas. El analista debe trabajar estrechamente con el usuario con el fin de obtener salidas que cumplan con sus necesidades. Existen seis puntos a observar en el desarrollo de salidas eficientes:

- *Diseñar una salida que satisfaga el objetivo planteado.*- La salida cuenta con un propósito que es definido durante el proceso de análisis junto con el usuario. No deben diseñarse salidas sin propósito específico, pues esto implica desperdicio de recursos en el desarrollo y en la operación.
- *Diseñar una salida que se adapte al usuario.*- Con base en entrevistas, observaciones y tal vez, prototipos, será posible crear salidas que se apeguen a la mayoría de las necesidades de los diferentes usuarios de un sistema.
- *Proveer la cantidad adecuada de información.*- Un sistema puede proveer de un gran volumen de información de salida, más no siempre, esto es necesario. El diseño de la cantidad de información que se muestra en una salida se puede evaluar basándose en las salidas que los procesos manuales actuales ofrecen, y basándose en la opinión de los usuarios. Lo anterior puede producir un mejoramiento en la eficiencia de las salidas que otorgue el sistema con respecto a los procesos o sistemas actuales. También es importante

tomar en cuenta la opinión del tomador de decisiones, quién normalmente, emplea poca información.

- *Asegurarse que la salida esté disponible donde se requiera.*- Para ser útil y aprovechable la salida debe de presentarse al usuario adecuado. El empleo de reportes que se despliegan en pantalla es una forma de alcanzar este objetivo en un sistema cliente-servidor, debido a que la información se encuentra disponible en el lugar en que cada usuario la necesita. No importa qué tan bien estén diseñados los reportes; si el usuario que requiere de ellos no los ve, el sistema perderá funcionalidad.
- *Proporcionar oportunamente la salida.*- Muchos reportes se requieren por día, por mes, por año o por algún otro período o por excepción. La presentación oportuna de los reportes es importante en el funcionamiento de los procesos de una empresa. La presentación a tiempo de los reportes, puede ser decisiva en la operación de la empresa.
- *Elección del método de salida.*- Puede ser en forma de reporte en papel, salida en pantalla, archivos, multimedia, entre otros. En un sistema cliente-servidor como “SAPA”, la salida en su mayoría es de tipo pantallas en línea, que a su vez generan reportes en línea, esto facilita la explotación de la información por parte de los usuarios.

A continuación se presenta el diseño de una salida de “SAPA” (Este diseño pertenece a la salida de un proceso llamado “Elaborar reporte 1”).

Sistema Sistema de Automatización del Proceso de Auditoría(SAPA)
Salida Reporte 1 (Resumen de órdenes de auditoría)

Objetivo

Generar un reporte impreso o en pantalla a través de quick report que nos entregue un resumen de la órdenes de auditoría.

Usuario Jefe de departamento de supervisión y auditoría

Procesos involucrados	Módulo	Frecuencia	Volumen
8.1 Elaborar reporte 1	8. Reportes	Eventual	5 por año

Datos	Tipo	Longitud	Rep
ENCABEZADO			
Año	Date	4	N
Número de orden de auditoría	Carácter	7	N
Número progresivo de revisión	Carácter	7	N
Inicio de revisión	Fecha	10	N
Fin de revisión	Fecha	10	N
Número de oficio de informe	Carácter	8	N
Fecha de informe	Fecha	10	N
Área revisada	Carácter	80	N
Observaciones determinadas	Numérico	2	N
Observaciones no solventadas	Numérico	2	N

Comentarios

La información que despliega el reporte es sobre el total de las auditorías capturadas en los diferentes años. Para hacer más selectiva la información que se consulta, el reporte se podrá filtrar por medio de los siguientes criterios para obtener información más específica:

- Número de Oficio.
- Número progresivo de revisión.
- Número de orden de auditoría.
- Área.
- Tipo de auditoría.
- Período de auditoría.
- Si es nueva auditoría o de seguimiento.

Para mostrar las observaciones determinadas sólo deberá realizarse una consulta de cuántas observaciones fueron creadas por auditoría, pero en el caso de las observaciones no solventadas deberá realizarse una consulta que filtre sólo aquellas observaciones que ya recibieron un seguimiento y que no han sido solventadas al momento de generar el reporte.

Pantalla

Print Preview

Resumen de Ordenes de Auditoria

AÑO	Número de Orden de Auditoria	Número Progreso de Revisión	Tipo de Auditoria	Periodo de Revisión Inicial	Periodo de Revisión Final	Número de Orden de Informe	Fecha Informe	Área Revisada	Cu Terminado (Si Terminó)	Cu Terminado (Si no Terminó)
1996	10/1996	21/1996	Reporte Masivo	1/1/94	5/1/94	NO O FCD 1	3/2/96	Director General Financiera Branded	1	3
1996	07/1996	13/1996	Reporte Masivo			212/1996		Secretaria Técnica	6	6
1996	03/1996	03/1996	Frangible			206/1996		Director General Recursos Humanos y Desarrollo	2	0
1996	19/1996	14/1996	Publicación de Pagos			362/96		Director General Financiera Branded	2	2
1997	10/1997	07/1997	Frangible			200/1997		Director General Recursos Humanos y Desarrollo	2	0
1997	11/1997	11/1997	Reporte Masivo			196/1997		Director General Recursos Humanos y Desarrollo	1	0
1997	14/1997	04/1997	Frangible			027/1998		Director General Recursos Humanos y Desarrollo	4	1
1998	05/1998	01/1998	Frangible			248/1998		Director General Recursos Humanos y Desarrollo	3	1
1998	06/1998	12/1998	Del Desempeño			411/1998		Director General Asesoría	1	1
1998	10/1998	13/1998	Reporte Masivo			701/1998		Director General Asesoría	2	2
1998	21/1998	08/1998	Reporte de Informes y Reportes			680/1998		Director General Recursos Humanos y Desarrollo	1	1
1998	24/1998	15/1998	Reporte Masivo			697/1998		Director General Asesoría	4	2
1998	30/1998	14/1998	Reporte Masivo			010/1999		Director General Inspección	2	1
1998	33/1998	16/1998	Reporte Masivo			048/1999		Secretaria Técnica	3	3
1998	34/1998	03/1998	Frangible			061/1999		Director General Recursos Humanos y Desarrollo	13	7
1999	03/1999	21/1999	Reporte Masivo			288/1999		Director General Asesoría	4	2
1999	12/1999	12/1999	Reporte Masivo			438/1999		Director General Recursos Humanos y Desarrollo	5	3

Página 1

Page 1 of 3

CAPÍTULO 5



*Desarrollo,
Documentación
y Pruebas*

5.1 Lineamientos del Desarrollo

La etapa de desarrollo es aquella en que se construye el sistema. Consiste en la programación de los módulos no existentes y la integración de éstos con los que se han de reutilizar de proyectos anteriores o con los que pertenecen a las herramientas que se van a emplear, siguiendo fielmente los diseños y estándares establecidos en la etapa de diseño. También se emplean como base los modelos de entidad relación (DER) y los diagramas de flujo de datos (DFD).

Los elementos que se desarrollaron en “SAPA” fueron los siguientes:

- *Interfaz de usuario.*- Consta de las pantallas y menús que emplea el usuario para procesar su información.
- *Procesamiento de datos.*- Rutinas que procesan los datos que introduce el usuario y que proporcionan resultados.
- *Almacenamiento.*- Lugar donde residen, tanto los datos introducidos por los usuarios, como los procesados.
- *Reportes.*- Son documentos que muestran información procesada por el sistema.

Para lograr construir los elementos anteriormente definidos, se requiere que los programadores involucrados en el proyecto los desarrollen por medio de un lenguaje de programación (en este caso *Delphi*).

El desarrollo del sistema se llevó a cabo de manera modular y cada programador fue responsable del desarrollo de ciertos módulos conforme a la planeación de actividades que se describió en la sección 4.1.2. Esta forma de trabajo requirió un estándar de programación, combinado con una buena organización de equipo de desarrollo. Los estándares y convenciones de programación ofrecen los siguientes beneficios específicos:

- Permiten "dar por hecho" ciertos aspectos de la programación. Se toma una decisión global en vez de una serie de decisiones locales. Por lo anterior, permite concentrarse en aspectos más importantes de la codificación.
- Permiten aprender fácilmente los aspectos conceptuales de un proyecto en lugar de distraerse en descifrar el código de cada programador. Se debe trabajar con un conjunto consistente de código.
- Reduce la proliferación de vicios de programación. Sin un estándar en nomenclatura es muy fácil que se nombre a lo mismo con dos nombres totalmente distintos. Esto puede ser extremadamente confuso para un programador que intenta dar mantenimiento al código de otros.
- Son una forma de compensar las debilidades del lenguaje de programación. Las convenciones nos pueden ayudar a diferenciar entre los varios tipos de entidades en un programa. Permiten distinguir si una variable es local, modular o global.
- Permiten enfatizar relaciones entre los datos y otros elementos del programa. Además, permiten también agrupar, por medio de nombres, una serie de estructuras de datos y las

funciones que las manipulan en conjuntos lógicos o grupos funcionales fáciles de asimilar conceptualmente.

- Transfieren conocimiento de un proyecto a otro. Un programador familiarizado con una disciplina de programación puede migrar de un proyecto a otro sin que el costo de aprendizaje sea elevado.

La consideración más importante en el nombramiento de entidades de un programa (constantes, variables, objetos, funciones, archivos, etc.) es que el nombre describa con precisión a la entidad que representa. Un nombre bien asignado generalmente se refiere al qué y no al cómo. Una estructura o registro que contiene información sobre un empleado podría llamarse “DatosEmpleado” o “RegistroDeEntrada”; el primero se refiere al qué y es lo correcto. El segundo se refiere al cómo y no dice nada acerca de la entidad que representa.

En el desarrollo de aplicaciones es muy importante el adoptar una nomenclatura que al menos cumpla con los siguientes requisitos:

- Los nombres deben ser suficientemente descriptivos, sin llegar a ser demasiado largos.
- Los nombres deben referirse al qué y no al cómo.
- Los nombres deben proporcionar información sobre el tipo de entidad del programa y el contexto en el que se encuentran.
- La nomenclatura debe ser consistente y homogénea a través de un programa, además de sencilla y expresiva para facilitar que un programador la adopte como disciplina.

A continuación se muestran los estándares de programación que se emplearon para el desarrollo de “SAPA”:

5.1.1 Declaración de Constantes

La convención seguida en el nombramiento de constantes es que su nombre se forma de un conjunto de palabras clave en mayúscula y separadas por guiones bajos (underscores). Las palabras claves se abrevian, pero sin sacrificar la claridad.

Formato:

<Prefijo><_><Palabra Clave1><_><Palabra Clave N>

Donde:

Prefijo.- Indica el tipo de constante que se trata. Se forma por tres letras en mayúsculas. Los caracteres válidos son definidos en la tabla que se muestra más adelante.

Palabra Clave1.- Alguna clave definida por el usuario representativa de la utilización de la constante. Son escritas en mayúsculas. Esta palabra clave es obligatoria.

Palabra ClaveN.- Igual que Palabra Clave1. Sin embargo, subsecuentes palabras clave son opcionales y se crean tantas como es necesario. Se aceptan abreviaciones mientras éstas sean lo suficientemente claras.

Prefijo	Tipo de Constante
INT	Numérica (excepto valores mínimos, máximos y longitudes)
STR	Cadena de caracteres (excepto mensajes de error)
MSG	Mensaje de programación (informativos, preguntas, de errores, etc)
ERR	Cadena de mensajes de error
MIN	Valor mínimo entre un rango de valores
MAX	Valor máximo entre un rango de valores
TBL	Tabla de una base de datos
CMP	Campo de una tabla

Tabla 5.1 Tipo constantes

Ejemplos:

```

Const MAX_COLUMNS = 50
Const MAX_OPEN_TABLES = 10
Const INT_DEBUG_LEVEL = 3
Const INT_VERSION_NUMBER = 3.5
Const STR_LOGO_FILE_NAME = "IMAGEN.BMP"
Const ERR_FILE_NOT_OPEN = "El Archivo no ha sido abierto"
Const ERR_FILE_NO_MEMORY = "No hay suficiente memoria en el sistema"
Const INT_NEW_LINE_CHAR = 10
Const INT_RETURN_CHAR = 13
Const MAX_EMPLOYEE_LEE = 32
    
```

5.1.2 Declaración de Variables

En el nombramiento de variables se utilizó lo que se conoce como notación Húngara. La notación Húngara fue definida por Charles Simonyi principalmente para su uso en el lenguaje C.

El nombre de una variable se compuso de dos o más elementos. Los primeros dos son obligatorios, el resto opcional. No se permitieron los guiones de piso (underscores) entre elemento y elemento del nombre.

Formato:

<Prefijo><Palabra Clave1 ><Palabra ClaveN><Calificador>

Donde:

Prefijo.- Indica el tipo de variable de que se trata y posiblemente su contexto. Este se forma de 1 a 3 letras en minúsculas y se construye de la siguiente manera:

Formato:

<Contexto><Arreglo><Tipo>

Donde:

<Contexto>: Indica el contexto de la variable. Es obligatorio excepto para variables locales. Los caracteres válidos son:

g: Indica que la variable es global al proyecto.

m: Indica que la variable es global al módulo o forma.

v: La variable es un parámetro pasado por valor.

r: La variable es un parámetro pasado por referencia.

s: La variable es declarada como local estática.

p: La variable es declarada como pública, es decir, que esta variable pertenece a un módulo o forma, pero puede ser accesada desde cualquier punto de la aplicación.

<Arreglo>: Indica si la variable es un arreglo, en cuyo caso se utiliza la letra ‘a’ para especificarlo.

Ejemplos: **gaiEdades** Arreglo global de edades enteras.

viGrado Parámetro por valor entero.

mbBandera Variable lógica global al módulo.

Palabra Clave1.- Alguna palabra clave definida por el usuario, que es representativa de la utilización de la variable. La primera letra es mayúscula y el resto en minúsculas. Esta palabra clave es obligatoria.

Palabra ClaveN.- Igual que Palabra Clave1. Sin embargo, subsecuentes palabras claves son opcionales y existen tantas como es necesario. La primera letra es mayúscula y el resto, minúsculas. Se aceptan abreviaciones mientras éstas sean lo suficientemente claras.

Calificador.- El calificador es usado para denotar derivados de un nombre base. Los calificadores son opcionales, pero no existe más de uno en un nombre de objeto. La primera letra es mayúscula y el resto en minúsculas. Se aceptan abreviaciones mientras estas sean lo suficientemente claras. Ejemplos:

miClaseFirst

miClaseLast

Ejemplos:

sNomAuditor String local a una función que contiene el nombre de un auditor.

bSeguimiento Variable local Boolean que indica nueva auditoría o seguimiento.

msNomArchivo String Global al módulo que contiene el nombre de un archivo.

viTotal Parámetro long pasado por valor que contiene un Total.

vvEstilo Parámetro tipo Variant pasado por valor.

IPaglnas Entero local que indica un número de páginas.

Otras especificaciones para las variables:

- Cuando se trata de declaraciones de variables locales a un procedimiento, éstas se declaran a continuación de la función dejando una línea en blanco e indentando las declaraciones con un tabulador de 2 posiciones. Después de la sección de declaración de variables se deja otro renglón en blanco y, a partir de ahí, se ponen las líneas de código subsiguientes.

- Nunca se declaran variables entre los bloques de código, siempre al inicio de la subrutina o función.
- La declaración de variables del mismo tipo es una por línea, agregando un comentario del lado derecho indicando el uso que se le asigna a ésta.
- Nótese que el uso de dígitos en nombres de variables no se permite en ningún caso.

Los nombres de campos dentro de tipos definidos por el usuario son congruentes con el nombramiento de variables.

5.1.3 Declaración de Tipos

Para el nombramiento de tipos definidos por el usuario (estructuras de datos o registros) se utilizaron las letras minúsculas, no permitiéndose el uso de guiones bajos (underscores), además de que tuvieron el prefijo **ty**.

Formato:

<Prefijo><Palabra Clave1><Palabra ClaveN>

Donde:

Prefijo.- Indica que se trata de un tipo definido por el usuario, y éste siempre fue **ty**.

Palabra Clave1.- Alguna palabra clave definida por el usuario, que es representativa de la utilización del tipo. La primera letra es mayúscula y el resto en minúsculas. Esta palabra clave es obligatoria.

Palabra ClaveN.- Igual que Palabra Clave1. Sin embargo, subsecuentes palabras clave son opcionales y existen tantas como es necesario. La primera letra es mayúscula y el resto en minúsculas. Se aceptan abreviaciones mientras éstas sean lo suficientemente claras.

La indentación de las declaraciones de las variables es un tabulador de 2 posiciones. La declaración de variables del mismo tipo es una por línea, agregando un comentario del lado derecho indicando el uso que se le asigna a ésta.

Ejemplo:

```
Type tyCiudad
  iCdCve   : Integer      'Clave de la Ciudad
  sCdDesc  : String      'Descripción de la Ciudad
  yCdIva   : Byte        'VA aplicable en la Ciudad
End type
```

5.1.4 Declaración de Subrutinas

El nombre de una subrutina se compuso de dos o más elementos. Los primeros dos obligatorios, el resto opcional. El primero corresponde a un verbo, el segundo al objeto sobre el cual se ejecuta la acción. Subsecuentes palabras pueden extender el concepto. Se utilizaron las letras minúsculas, sin hacer uso de guiones bajos (underscores) entre elemento y elemento del nombre.

Formato:

<Verbo> {<Objeto1 > I <Palabra Clave1 >} {<ObjetoN> I <Palabra ClaveN>}

Donde:

Verbo.- Indica la acción a ejecutar.

Objeto1 ó Palabra Clave1 Algún objeto o palabra clave definida por el usuario, que es representativo(a) de la utilización de la subrutina. La primera letra es mayúscula y el resto en minúsculas.

ObjetoN ó Palabra ClaveN Subsecuentes objetos o palabras clave son opcionales y existen tanto(a)s como son necesario(a)s. La primera letra es mayúscula y el resto en minúsculas. Se aceptan abreviaciones mientras éstas sean lo suficientemente claras.

Una subrutina se utiliza cuando se requiere ejecutar una serie de instrucciones, que no retornan un valor.

Ejemplos:

```
procedure TFrmOrdenAudtria.ConsultaOrden(viSumaPagada: integer);
procedure TFrmOrdenAudtria.FormatGrid;
procedure TFrmOrdenAudtria.PresentaResultado;
```

5.1.5 Declaración de Funciones

El nombre de funciones, al igual que el de las subrutinas, se compuso de dos o más elementos. Los primeros dos obligatorios, el resto opcional. El primero corresponde a un verbo y el segundo al objeto sobre el cual se ejecuta la acción. Subsecuentes palabras extienden el concepto. Se utilizaron las letras minúsculas, sin hacer uso de guiones bajos (underscores) entre elemento y elemento del nombre.

Formato:

<Verbo> {<Objeto01> I <Palabra Clave1>}{ <ObjetoN> I <Palabra ClaveN>}

Donde:

Verbo	Indica la acción a ejecutar.
Objeto1 ó Palabra Clave1	Algún objeto o palabra clave definida por el usuario, que es representativo(a) de la utilización de la función. La primera letra es mayúscula y el resto en minúsculas.
ObjetoN ó Palabra ClaveN	Subsecuentes objetos o palabras clave son opcionales y existen tantos(as) como son necesarios. Se aceptan abreviaciones mientras éstas sean lo suficientemente claras.

Una función se utilizó cuando se requirió ejecutar una serie de instrucciones que retornan un valor.

Ejemplos:

```
Function TfrmMenu.ValidaHora(vsTextoValidar: string): string;  
Function ObtenCiudades: Integer;  
Function ObtenSalario: Double;
```

5.1.6 Comentarios

La apreciación en el uso de comentarios generalmente nace de programadores que se han dedicado al mantenimiento de código de otros y saben de la gran ayuda que puede tener un código bien estructurado y documentado. Así, se muestran a continuación las convenciones sobre la utilización comentarios en los programas:

Encabezado por Módulo y/o Forma:

Cada forma y cada módulo debe contener en la sección "Declaraciones Generales" el siguiente comentario:

```
'Lenguaje:  
' Nombre y versión del lenguaje a utilizar por el módulo  
' Ejemplo: Borland Delphi 5.0 for Windows  
'Proyecto:  
' Nombre del Sistema  
' Ejemplo: Sistema de Automatización del proceso de auditoría ("SAPA")  
'Responsable(s): (incluyendo la fecha de última modificación)  
' Ejemplo. José Antonio Cejudo Ibarra 14/01/2003  
'Descripción.  
' Escribir aquí las líneas suficientes para exponer claramente la funcionalidad del módulo.  
'Tablas utilizadas:  
' Nombrar las tablas que se utilizan en alguna parte del módulo o forma  
'Stored Procedures utilizados:  
' Citar los nombres de los stored procedures que son utilizados en alguna parte del módulo o  
' forma  
'Observaciones  
' Algún comentario importante con relación a la funcionalidad del módulo o la forma  
'Constantes  
'Tipos  
'Variables
```

Encabezado por Subrutina y/o Función:

Todas las funciones y subrutinas se inician con el siguiente comentario:

```

' Descripción:
'   Breve descripción de la función de la subrutina o función del módulo.
' Parámetros:
'   Aquí se describen por qué son utilizados cada uno de los parámetros.
' Stored Procedures:
'   Si la subrutina o función hace uso de stored-procedures, se debe hacer una breve descripción de
'   estos. Si no, no debe estar presente esta sección.,
' Tablas utilizadas:
'   Enumerar las tablas que se utilizan en la subrutina o función.
' Variables globales que modifica:
'   Aquí se indica si la subrutina o función hace modificaciones a alguna(s) variable(s) global(es) , en
'   qué casos y por qué.
' Valores de retorno (constantes definidas cuando es diferente de True o False):
'   Para el caso de funciones se debe indicar el tipo de valores de retorno que la función puede
'   regresar. Para subrutinas, esta sección puede ser utilizada en el caso de pasar valores por
'   referencia, de lo contrario, no debe existir.
' Observaciones:
'   Si hay algún aspecto importante o detalle sobre el funcionamiento de la subrutina o función, que
'   requiera ser aclarado, se define como nota en esta sección.
    
```

Comentarios en las líneas de Código:

- Se incluyeron comentarios en el código, con el fin de tener ayudas extras en el mantenimiento de los programas. No se pretendió comentar cada línea de código porque hubiera resultado contraproducente, sino por bloques de líneas, explicando el por qué se hizo algo, sobre todo cuando el propio código no fue explicativo por sí mismo.
- Se recalcó, por medio de conceptos cortos, cuando cierta parte del código dependió de objetos externos (no parte del módulo), como son tablas, triggers, stored-procedures, otras formas, etc., u otros aspectos, como por ejemplo cuando una función asumió un cierto formato de fechas, un rango de valores u otras dependencias.
- En los casos en que se comentó por línea, el comentario debió ir a la derecha de la línea de código.
- Cuando se comentó por grupos de líneas, se escribió el comentario en una línea arriba del bloque de código.
- Cuando fueron bloques de código grandes o se buscó resaltar la funcionalidad, se puso el comentario de la siguiente forma:

```

'*****
'Comentario de una o varias líneas
'*****
    
```

5.2 Desarrollo de módulos

5.2.1 Módulos

5.2.1.1 PACA

El desarrollo del módulo de “PACA” fue uno de los más complejos. Implicó el manejo de una relación maestro detalle entre las tablas “PACA”, “Subtitulo_PACA” y “Descripcion_PACA”.

El objetivo del módulo es permitir al usuario la captura del programa anual de control y auditoría, que se realiza año con año en el área de contraloría de cualquier dependencia de gobierno. El PACA contiene todas las revisiones (auditorías) que se llevarán a cabo durante un año.

El problema más difícil de resolver fue el lograr integrar el manejo en línea de los registros de las tres tablas con relación maestro detalle para proporcionar una familiaridad al usuario con respecto al documento PACA, y no forzarlo a desplegar diferentes pantallas para cada tabla. Es necesario tener una revisión válida en la tabla “PACA” para poder capturar múltiples rubros pertenecientes a esa revisión. A su vez, es necesario tener un rubro válido capturado en la tabla “Subtitulo_PACA” para poder capturar diferentes descripciones.

El problema descrito, se resolvió por medio de un registro temporal en la tabla “PACA”, que se hace definitivo en el momento de oprimir el botón **Guardar**. Esto permite la captura de los rubros y descripciones que están directamente relacionados con la revisión que se pretende dar de alta. A través de validaciones activas en el momento de la captura, se logró evitar problemas de inconsistencia de datos almacenados al no permitir la captura de un **Rubro** sin previa captura de una revisión, ni la captura de una **Descripción** sin previa captura y alta de algún **Rubro**.

Este módulo es, en esencia, un módulo de consulta, porque la captura del PACA sólo se lleva a cabo una vez al año. Su utilidad principal radica en la capacidad de consultar las revisiones en cualquier época el año. Para facilitar la consulta se implementaron botones de navegación por **revisión**, **rubros** y **descripciones**.

La otra función de que consta el módulo es la de generar los documentos oficiales PACA resumen y PACA detallado. Ambos reportes se generan en Excel y fue necesario hechar mano de la tecnología OLE (*Object Linking and Embedding* [Vinculación e incrustación de objetos], empleada para manipular objetos de aplicaciones como Word, Excel o Power Point), para poder generar la interacción entre “SAPA” y Excel. Los datos contenidos en las tablas de la base de datos, se envían al documento (Plantilla del PACA en Excel) donde se ubican por medio del sistema de renglones y columnas que posee Excel.

El proceso anterior está oculto al usuario, pues si desea crear el Programa Anual de Control y Auditoría Resumido o el Programa Anual de Control de Auditoría Detallado, sólo deberá presionar el botón **Documento**.

Aunque parece trivial, uno de los problemas de programación más complejos se dio al tratar de desplegar la información en los documentos de Excel. Los principales factores que influyeron en esta complejidad fueron: el respetar el formato de los documentos oficiales proporcionados por el área de contraloría, el manejo del volumen de información a desplegar (el documento PACA detallado puede ser de mas de 100 páginas), y la detección por parte de “SAPA” del número de aplicaciones Excel que fue abriendo.

Los primeros dos problemas se resolvieron a través del uso de algoritmos que respetaron el formato de los querys hechos a la base de datos y el formato de los documentos oficiales. El tercer problema se resolvió por medio de un manejo de excepciones.

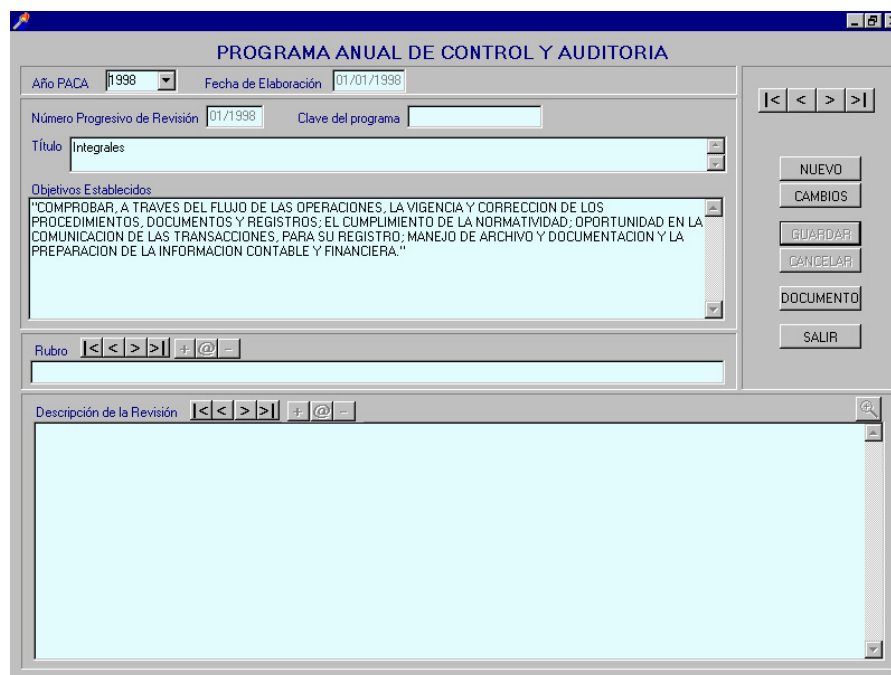


Figura 5.1 Pantalla PACA

5.2.1.2 Orden de auditoría

Al construir este módulo se presentaron diferentes problemáticas a resolver. Es aquí donde se define qué tipo de auditoría se va a realizar: nueva o de seguimiento. Por medio de un radio button se selecciona el tipo de auditoría a capturar. Esta opción permite: habilitar la funcionalidad de captura de observaciones a dar seguimiento (sólo válida para las auditorías de seguimiento), a través del módulo “**Consulta al Inventario de Observaciones**”; como

bandera para filtrar nuevas auditorías en el módulo “**Cédulas de observaciones**”; de seguimiento para el módulo “**Cédulas de seguimientos**”, y controla la generación de un documento de nueva auditoría o de seguimiento.

En este módulo se realizan consultas a las tablas “PACA”, “Clase”, “Áreas”, “Personal” y “Tipo_Auditor” para llenar los combos **No. Progresivo de revisión**, **Tipo de auditoría**, **Área**, **Nombre** y **Tipo** personal respectivamente.

La captura de participantes en el proceso de auditoría consistió, primero, en validar la captura completa de los datos de una auditoría y la alta de la misma por parte del usuario. Fue necesario llevar a cabo estas validaciones porque es necesario contar con una auditoría activa para poder llevar a cabo la captura de nuevos registros en la tabla “Participantes”. Al tener las validaciones mencionadas, se pensó en la manera de capturar, de manera práctica, los diferentes tipos de participantes de una auditoría, llegando a la idea de contar con un campo de selección de **Personal** y otro de **Tipo de personal** participante, además de contar con un botón para agregar el participante (+) y otro para eliminar un participante(-). La consulta de los participantes en una auditoría se decidió hacer por medio de un grid, que refleja, en tiempo real, la alta o baja de algún participante en el documento Orden de Auditoría.

Una observación importante es que este proceso de alta y baja de participantes es de vital importancia en el manejo de la seguridad del sistema por lo siguiente: los auditores del área de contraloría (exceptuando a los de nivel igual o superior al jefe de departamento) sólo pueden consultar las auditorías que ellos mismos han creado, es decir, las auditorías en las cuales aparecen como responsables. Las demás auditorías quedan ocultas. Tomando en cuenta que es en este módulo donde se tiene que definir los participantes de la auditoría y que, precisamente, uno de los participantes es el o los responsables de la auditoría, fue necesario validar la captura obligatoria de al menos un responsable de la auditoría que se está dando de alta.

La información de los responsables de las auditorías es lo que permite filtrar la información de las auditorías por auditor. Por ejemplo: si Juan Pérez es dado de alta como único responsable en las auditorías 01/2003, 05/2003 y 07/2003, él será el único capaz de consultar y modificar la información de esas auditorías. Esto permitió dar confidencialidad a la información de cada auditor.

Para fines de administración fue necesario crear también un estatus que permitiera a cierto tipo de usuario la consulta del total de auditorías contenidas en el sistema. Este **estatus** de usuario fue denominado “**ADMIN**” y se puede asignar a cualquier usuario a criterio del Director de Supervisión y Auditoría.

La parte en la que el módulo genera los documentos Orden de Auditoría nueva o de seguimiento, implicó un grado de complejidad tal que, prácticamente, puede ser considerado como el desarrollo de otro módulo de consulta de órdenes de auditoría por medio de un documento en Word. La interacción entre la “*SAPA*” y Word se desarrolló con la tecnología OLE (*Object Linking and Embedding* [Vinculación e incrustación de objetos]), que sirven para manipular objetos de aplicaciones como Word, Excel o Power Point. Por medio de un objeto OLE de tipo Word, plantillas de los documentos y stored procedures de consulta se construyó

esta parte del módulo: primero se realiza la consulta a la base de datos con datos por medio de *Delphi* y el *stored procedure*. A continuación se abre la aplicación Word con el objeto OLE y por medio del objeto y de marcadores nombrados (simulando campos) en la plantilla, se vacía la información de la consulta en el documento.

Tipo_Audidores	Nombre	Cargo
Atencion_orden	Lic. Juan Rodríguez Díaz	Director de Supervisión y Auditoría
Copia_orden	C.P. Eduardo Rafael Cruz León	Jefe de Departamento de Auditoría
Responsable	C.P. Eduardo Rafael Cruz León	Jefe de Departamento de Auditoría

Figura 5.2 Pantalla Orden de Auditoría

5.2.1.3 Consulta del inventario de observaciones

Una de las principales razones para el desarrollo de este sistema fue la necesidad de contar con un inventario de observaciones automatizado, es decir, que permitiera al usuario la administración y consulta de las observaciones de una manera rápida y sencilla. Precisamente, esta es la función de este módulo.

En esencia, es un módulo de consulta que interactúa con el módulo “**Orden de auditoría**” para poder generar las órdenes de auditoría de seguimiento. La información que muestra consiste en el total de las observaciones pendientes de seguimiento pertenecientes a todas las auditorías realizadas en años anteriores o, inclusive, en el mismo año, y que se encuentran almacenadas en la tabla “Observaciones”.

Para facilitar al usuario la consulta del inventario se implementaron dos filtros para las observaciones: uno por **clase de auditoría** y otro por **área a auditar**. Es importante notar que

ésta es una de las pantallas que proporciona mayor utilidad al usuario porque le evita perder tiempo de consulta del inventario de observaciones físico, en el momento en el que está creando una orden de auditoría de seguimiento.

Al mismo tiempo que se integran una a una las observaciones a las que se desea dar seguimiento, se valida que el sistema permita ver estas observaciones para darles seguimiento en el módulo “Cédula de Seguimientos”.

Por último cabe mencionar que este módulo también posee el desarrollo de la funcionalidad de navegación entre las diferentes observaciones que sean consultadas.

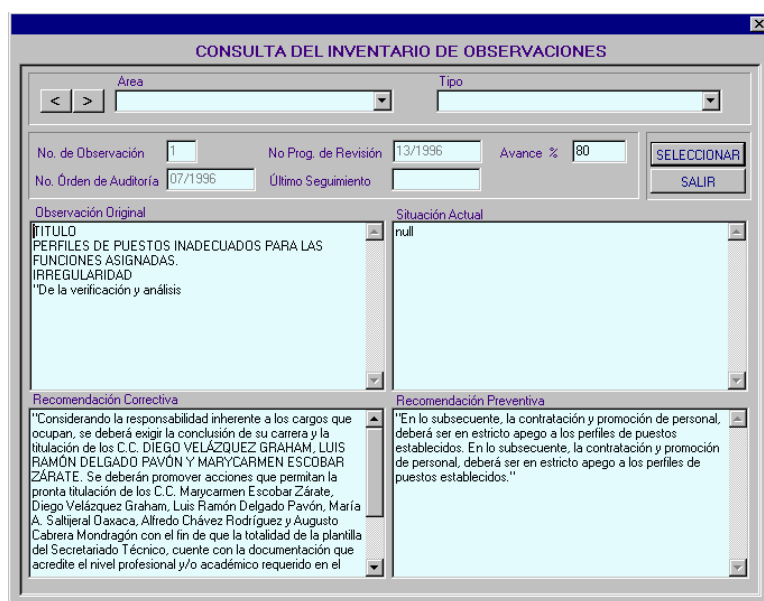


Figura 5.3 Pantalla Consulta al Inventario de Observaciones

5.2.1.4 Acta de inicio

Este módulo fue uno de los más sencillos a desarrollar porque su funcionalidad consiste, en esencia, en actualizar los datos de la tabla “Auditoría” que en el momento de dar de alta una orden de auditoría adquieren el valor nulo. Se dividió una parte de sólo consulta y una parte de altas bajas, cambios y consulta de Actas de inicio.

También cuenta con la funcionalidad de agregar o borrar participantes en la auditoría de tipo testigo y servidor auditado. Para lograr lo anterior fue necesario que se crearan primero consultas a las tablas “Personal” y “Tipo_personal” para llenar los combos **Nombre** y **Tipo personal**, respectivamente. Después se implementaron botones que insertan y borran registros de la tabla “Participantes”.

Finalmente se decidió desplegar una consulta de los tipos de participantes agregados al Acta de inicio en un grid. La consulta se actualiza cada vez que se agrega o borra algún participante en la auditoría a través del módulo “Acta de Inicio”.

Al igual que los módulos anteriores desde aquí se puede generar el documento “Acta de Inicio” al oprimir el botón **Documento**. La forma de manejar la interacción entre “SAPA” y Word es la misma que en la “Orden de auditoría” y en general es la misma para todo el sistema.

Nombre	Tipo_Audidores	Cargo
Lic. Augusto Cabrera Mondragón	Servidor_Auditado	Subdirector de Comunicación Social y Relaciones Públicas
Lic. Antonio Eliceo Mora Telléz	Testigo	Director General de Planeación Financiera

Figura 5.4 Pantalla Acta de Inicio

5.2.1.5 Carta planeación

La funcionalidad del módulo de “Carta planeación” es muy similar a la funcionalidad de “Acta de inicio”. La principal diferencia consiste en que los campos actualizados en la tabla “Auditoría” son diferentes a los actualizados en “Acta de inicio”.

Otra diferencia mínima radica en la manera de manejar el combo **Trimestre** con datos fijos en lugar de ser llenado a través de una consulta. Por lo demás, el módulo se desarrolló de manera idéntica al de “Acta de inicio”. Aunque la cantidad de datos capturados para la “Carta planeación” son pocos, es en este módulo donde se puede ver la optimización del uso de la información de manera más clara: no es necesario volver a capturar información ya capturada en “Orden de auditoría” o en “Acta de inicio”, y sólo se consulta desde la base de datos para generar el documento Carta planeación completo.

5.2.1.6 Cronograma de actividades

El módulo de “**Cronograma de actividades**” es el más sencillo junto con el de “**Evaluación de auditoría**”. Su principal función consiste en desplegar una pantalla en la cual se introduce el número de orden de auditoría y partir de este número de orden de auditoría se desarrolló una consulta a la tabla “Auditoría” desde donde se traen los datos necesarios para llenar ciertos campos de una plantilla en Excel del Cronograma de actividades.



Figura 5.5 Pantalla Carta Planeación

5.2.1.7 Marco conceptual

La función del módulo es la de crear diferentes documentos Marco Conceptual para cada auditoría que se realiza. Prácticamente este módulo opera de la misma forma que los módulos “**Acta de inicio**” y “**Carta planeación**”, sólo que en lugar de programarse actualizaciones en la tabla “Auditoría” se manejó una relación maestro detalle entre la tabla “Auditoría” y la tabla “Cedula_Marco”.

Es en este módulo donde se presentaron las mayores modificaciones al diseño que originalmente se definió, llegando inclusive a afectar la base de datos. La causa de esta contingencia fue la complejidad con la que se pretendió construir la pantalla: inicialmente se propuso que desde esta pantalla el usuario pudiera manipular las descripciones de ciertos rubros contenidos en el módulo “**PACA**” (finalmente el campo **Objetivo** de “**Marco conceptual**” contiene estas descripciones de manera idéntica a las del “**PACA**”).

Esta situación llevó al analista a pensar que era mucho más fácil manejar las descripciones ya capturadas en lugar de volverlas a teclear; además, durante las reuniones de análisis tampoco quedó claro que existen varios Marcos Conceptuales por auditoría. Lo anterior propició, que al mostrar al usuario la pantalla, el manejo fue mucho más complejo de lo que esperaba y no cumplía con los requerimientos, por lo que fue necesario reconstruir el diseño de la pantalla y crear la tabla “Cedula_Marco” como detalle de la tabla “Auditoría” (para cada auditoría existen muchos Marcos Conceptuales).

El nuevo diseño dividió el funcionamiento de la pantalla en dos partes. Primero, se desarrolló la parte de consulta de auditorías, que consta de un encabezado con campos de consulta y botones de navegación sobre los registro de la tabla Auditoría. La finalidad de esta parte del módulo fue permitir la selección de la orden de auditoría de la cual se quiere manipular sus Marcos Conceptuales.

La segunda parte del módulo permite dar de alta y hacer cambios en la tabla “Cedula_Marco” en la cual se almacenan los Marcos conceptuales. Es aquí también donde se implementaron botones de navegación para poder consultar los diferentes Marcos conceptuales para cada auditoría (no fue necesario crear un botón **Primero** y **Último** porque el número de Marcos Conceptuales por auditoría es menor a 10 normalmente).

En este módulo se implementaron los botones **Zoom**, que maximizan ciertos campos permitiendo una mayor facilidad de edición.

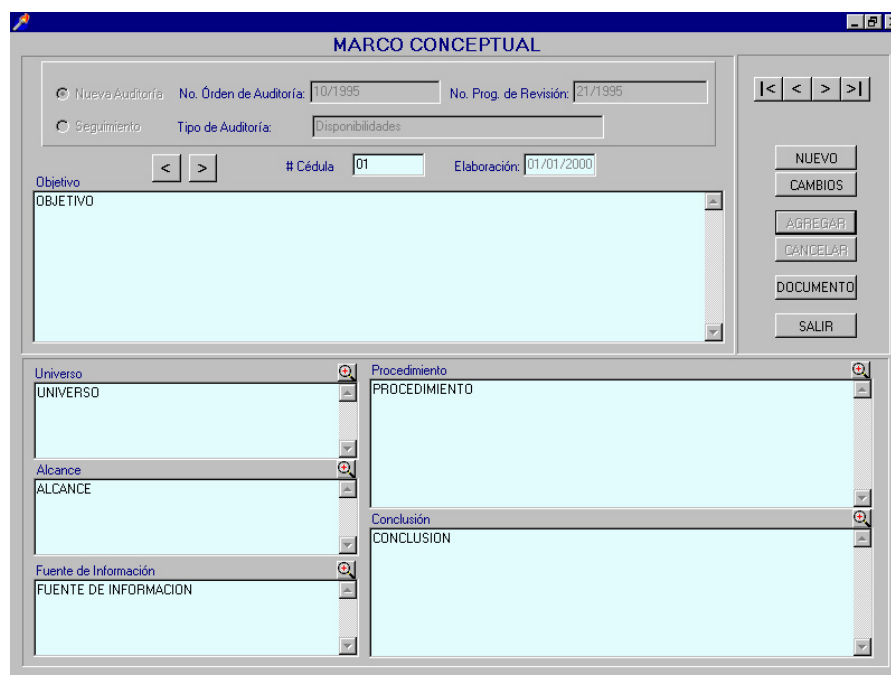


Figura 5.6 Pantalla Marco Conceptual

5.2.1.8 Cédula de observaciones

Este es uno de los módulos más importantes del sistema. En este módulo se lleva a cabo la captura de las observaciones para cada auditoría, y por lo tanto es donde se ingresan las observaciones al inventario de observaciones. Según el proceso de “Generación de Observaciones en una Auditoría” (revisado con anterioridad), el módulo debía de contar con una forma de generar inicialmente las observaciones totales de una auditoría, para después, borrar algunas dependiendo de la revisión realizada por el Director de Supervisión y Auditoría y la discusión con el área auditada (la eliminación de observaciones implica la reenumeración consecutiva de las observaciones restantes). Además, este proceso debía llevarse a cabo antes de la integración de las observaciones, al inventario.

Para resolver las necesidades de operación de este módulo, primero se implementó una sección de consulta de los registros de las auditorías permitiendo al usuario localizar la auditoría para la que quiere generar observaciones. Esta sección dentro del módulo es idéntica a la descrita en “Marco Conceptual” (campos de consulta y botones de navegación en la parte superior del módulo).

A continuación se decidió implementar una sección de consulta, altas y cambios de observaciones por auditoría. Esta sección trabaja sobre las tablas “Observaciones” y “Cedula_Observaciones” (más adelante se describirá por qué el trabajo sobre dos tablas). La consulta de observaciones se implementó por medio de dos botones de navegación **Anterior** y **Siguiente** que recorren los registros de la tabla “Observaciones”. Debido a que el número de observaciones por auditoría, normalmente es menor a 10, se decidió no implementar los botones de navegación **Primero** y **Último**.

Figura 5.7 Pantalla Cédula de Observaciones

Para este módulo también se implementó, por medio de OLE, una sección de generación de documento “Cédula de observación”.

Otra funcionalidad implementada consistió en el agregar un estatus llamado **Procede** o **No Procede**, que permitió, la manipulación de la información de las observaciones sin que estas fueran integradas al inventario de observaciones. Mientras este estatus permanezca sin ser activado para una observación esta no se integra al inventario de observaciones. Si se activa el estatus **No procede**, el sistema manda imprimir la observación seleccionada, la borra y vuelve a numerar de manera consecutiva las observaciones restantes en la auditoría.

La reenumeración de las observaciones emplea las tablas “Observaciones” y “Cedula_observaciones” que contienen información idéntica. El procedimiento se describe a continuación: se borran todas las observaciones de una auditoría en la tabla “Cedula_observaciones” y se vuelven a copiar desde la tabla “Observaciones”, aquellas observaciones de la misma auditoría que tengan estatus **Procede** o que permanezcan sin definir estatus alguno, con una nueva numeración. Por otro lado si se activa el estatus **Procede**, las observaciones se integran al inventario de observaciones pudiendo ser consultadas desde el módulo “**Consulta del inventario de observaciones**”, al mismo tiempo se deshabilita el botón **Cambios** para evitar futuras modificaciones de la observación capturada desde este módulo (los cambios que sea necesario hacer, tendrán que ser realizados desde el módulo “**Cédula de seguimientos**” durante una auditoría de seguimiento).

5.2.1.9 Cédula de seguimientos

Así como el módulo de “**Cédula de observaciones**” sirve como acceso al inventario de observaciones, este módulo sirve en esencia para evaluar el seguimiento de las observaciones de una auditoría de seguimiento y decidir en qué momento permanece o sale una observación del inventario.

La primera sección de este módulo implementa un encabezado de consulta de la tabla “Auditoría”, donde se puede navegar a través de los botones **Primero**, **Anterior**, **Siguiente** y **Último**. Esta sección es idéntica a la que poseen todos los módulos de la parte principal del sistema y tiene el mismo fin de ubicar la auditoría para la que se quiere generar una cédula de seguimiento.

La segunda sección del módulo es la que maneja las Cédulas de Seguimiento que se encuentran almacenadas en la tabla “Auditoria_seguimiento”. Las Cédulas de Seguimiento contienen observaciones a las que una Auditoría de Seguimiento da seguimiento, valga la redundancia. La auditoría de seguimiento integra las observaciones a las que da seguimiento en el módulo “**Orden de auditoría**” a través del módulo “**Consulta al inventario de observaciones**”. Al ser integradas las observaciones, se inserta un registro en la tabla “Auditoria_seguimiento” que relaciona cada observación con su auditoría de seguimiento correspondiente. Si en algún momento se requiere que alguna observación seleccionada no sea revisada por una auditoría, se implementó la funcionalidad de borrar estos registros que ligan auditorías con observaciones en la tabla “Auditoria_seguimiento”, que se activa por medio del

botón **Desligar**. Debido a que el número de observaciones a las que da seguimiento una auditoría regularmente no es mayor a 10, sólo se implementaron los botones **Anterior** y **Siguiente**.

La siguiente funcionalidad implementada en el módulo corresponde a la sección que evalúa el avance del seguimiento de una observación y permitiendo su permanencia o salida del inventario de observaciones. Mediante los campos **Avance** y **Solventada** se puede definir qué porcentaje de la observación ha sido resuelta o solventada por el área auditada, al mismo tiempo de manera automática se coloca una palomita en el campo **Solventada** en el momento de escribir un **Avance** igual al 100% y recíprocamente al seleccionar el campo **Solventada** se escribe el **Avance** de 100%.

Debido a que el porcentaje de solventación de una observación puede variar durante el desarrollo de la auditoría de seguimiento, fue necesario implementar el campo **Definitivo**. Este campo hace definitivo el porcentaje de solventación de una observación, lo que indica si permanece en el inventario (menor al 100%), o si sale (100%). Aquellas observaciones que poseen un porcentaje de solventación igual al 100% dejan de verse en el módulo “**Consulta al inventario de observaciones**”. Si su porcentaje es menor al 100%, se pueden seguir consultando en el módulo mencionado para un posterior seguimiento en otra auditoría de seguimiento. Al activar el campo **Definitivo** también se deshabilitan los botones **Cambios** y **Desligar**, quedando cerrado el acceso a los datos de la observación a la que se da seguimiento hasta que se dé seguimiento, de ser necesario, en una nueva auditoría.

Podemos mencionar que para este módulo, también se implementó por medio de OLE y *Delphi* (de igual manera que en los módulos anteriores), una sección en la que se genera el documento **Cédula de seguimientos**. Finalmente se implementó la funcionalidad de maximización de campos por medio del botón **Zoom**.

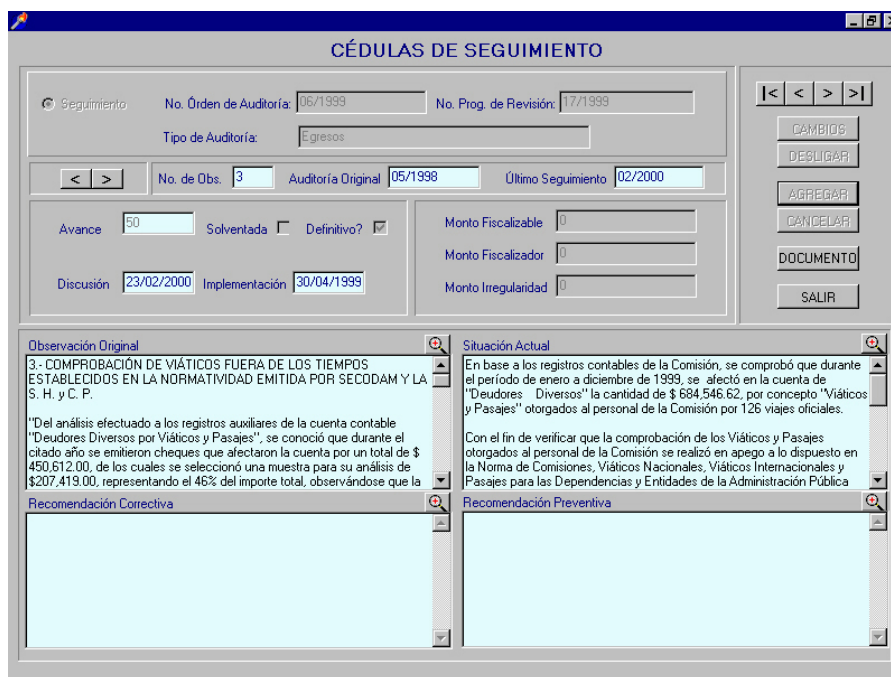


Figura 5.8 Pantalla Cédulas de Seguimiento

5.2.1.10 Informe de auditoría y cédula única de auditoría

La primera sección de este módulo es común a los demás módulos de la parte principal del sistema y consiste en una serie de campos de consulta de los principales datos de las auditorías contenidas en la tabla “Auditoria” con la posibilidad de navegar entre sus registros a través de los botones de navegación **Primero, Anterior, Siguiente** y **Ultimo**.

La segunda sección del módulo contiene una serie de campos que pertenecen también a la tabla “Auditoria” pero que pueden ser dados de alta, baja y cambios. En realidad, la implementación que manipula estos datos realiza actualizaciones sobre la tabla “Auditoria”.

La tercera sección del módulo consiste en un grid donde se despliegan las personas a quien se desea enviar copias del informe. Esta sección también consta de un combo de selección de personal y de dos botones, uno de agregar y otro para borrar personal de copias. Los datos que manejan el grid y los botones, están contenidos en la tabla “Participantes”. Los datos que se manejan en el combo **Nombre** son obtenidos del total de los registros la tabla “Personal”. La funcionalidad completa de esta sección se implementó para activarse sólo cuando se está ingresando un **Nuevo** informe o cuando se hacen **Cambios** sobre uno existente.

La cuarta sección de este módulo consta de dos campos contenidos en la tabla “Auditoria”. Los datos de estos campos pueden ser dados de alta baja o cambios con el resto de los campos de la sección dos del módulo. La razón de la independencia de los campos con respecto a las demás secciones, es porque son datos que pertenecen a otro documento que se genera en este mismo módulo: la Cédula Única de Auditoría. Inicialmente se considero crear un módulo independiente llamado “**Cédula Única**”, pero debido a que sólo se consideran dos campos para generar este documento, se optó por integrarlos dentro del módulo “**Informe de auditoría**”.

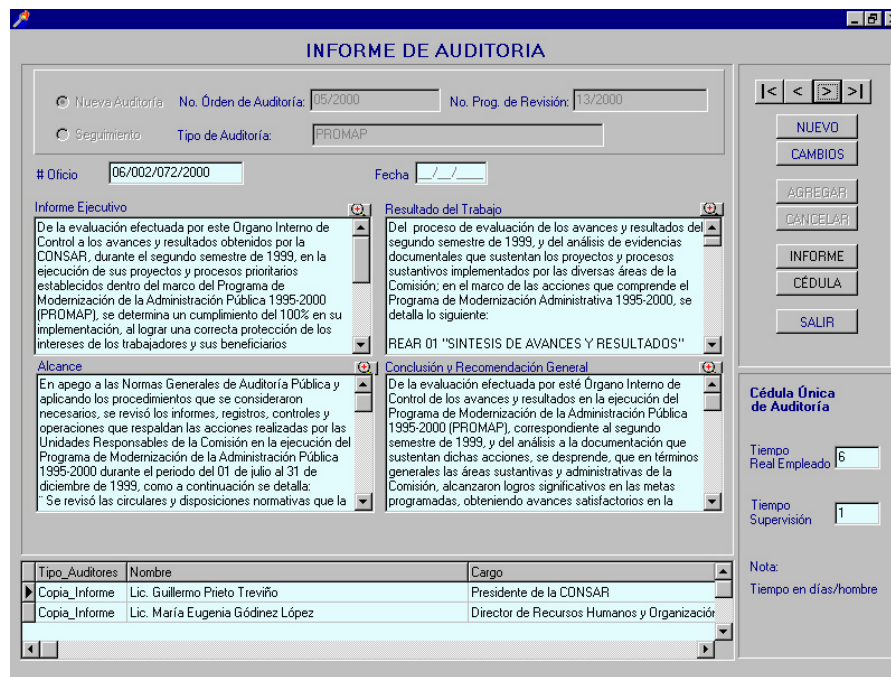


Figura 5.9 Pantalla Informe de Auditoría

Finalmente consideramos la sección donde se encuentran los botones **Informe** y **Cédula** que generan los documentos Informe de Auditoría y Cédula Única respectivamente.

La técnica empleada para la interacción entre “SAPA” y Word es la misma que se empleó en los demás módulos (OLE). Este módulo es el que más explota la información capturada en los módulos anteriores para la generación de documentos (El informe de Auditoría es el documento que contiene más datos provenientes de la tabla “Auditoria”).

5.2.1.10 Evaluación de auditoría

El módulo de “**Evaluación de auditoría**” es el más sencillo junto con el de “**Cronograma de actividades**”. Su principal función consiste en desplegar una pantalla en la cual se introduce el número de orden de auditoría y a partir de éste se desarrolla una consulta a la tabla “Auditoria” desde donde se traen los datos necesarios para llenar ciertos campos de una plantilla en Word del documento Evaluación de auditoría.

5.2.1.11 Catálogos

Este módulo se dividió en tres secciones controladas por una sección general o barra de herramientas. La barra de herramientas se implementó con dos grupos de botones. El primer grupo tiene funciones de navegación en los registros de cada uno de los módulos (Botones **Primero**, **Anterior**, **Siguiente** y **Último**) y el segundo grupo sirve para el manejo de cada registro (Botones **Nuevo**, **Cambios**, **Agregar** y **Borrar**).

The screenshot shows a window titled 'CATÁLOGOS' with a toolbar containing navigation and action icons. Below the toolbar are three tabs: 'Areas', 'Clases', and 'Personal'. The 'Personal' tab is active, displaying a form with the following fields:

Saludo	Nombre	Iniciales
Lic.	Fernando Joel Campuzano Cervantes	FJCC
Matrícula	RFC	Cargo
506	CACF550603HK	Director de Finanzas y Recursos Material
Credencial	Originario	Estado Civil
506	México D.F.	Soltero
Area	Edad	
Dirección de Finanzas y Recursos Materiales	45	
Domicilio		
Cerrada San Borja # 30, Col. Del Valle, Deleg. Benito Juárez, C.P. 0		

Figura 5.10 Pantalla Catálogos

Este es el único módulo en el que se decidió emplear los *Data Controls* de *Delphi* por la facilidad en su uso y la sencillez en la funcionalidad de cada una de las secciones del módulo. La facilidad de uso de los *DataControls* se ve a continuación: primeramente se crean las formas con los controles que contendrán los datos, luego, se crean objetos Ttable que contienen el total de los registros de la tabla a consultar (en este caso las tablas “Áreas”, “Clase” y “Personal”), finalmente, se ligan los controles de la(s) forma(s) a cada uno de los campos de los objetos Ttable. Para realizar la navegación se ejecutan eventos sobre los objetos Ttable que son invocados desde los eventos de los botones.

La primera de las tres secciones que controla la barra de herramientas y que está basada en el uso de los *DataControls*, es la sección que contiene la información de las Áreas de la dependencia de gobierno (consulta a la tabla “Áreas”).

La segunda sección contiene información referente a las clases de auditoría almacenada en la tabla “Clase”.

La última sección contiene los datos del personal de la dependencia de gobierno almacenados en la tabla “Personal”. En esta sección también se implementó una consulta a la tabla “Área” para llenar el combo de áreas a la que pertenece el personal.

5.2.1.12 Reportes

El módulo “**Reportes**” se divide en cuatro secciones: tipo de reporte, filtros para observaciones, filtros para auditorías y botones de control.

La implementación del funcionamiento del módulo consiste en una sección de selección de tipo de reporte a lanzar. A continuación se tienen opciones de filtrado por observaciones o auditorías (estas opciones se habilitan dependiendo del tipo de reporte elegido). Finalmente se implementó una sección de botones de control que sirve para la generación de los reportes y salir.

La primera sección consiste de cuatro opciones de reporte:

- **Resumen de órdenes de auditorías** que genera una consulta a la tabla “Auditoria”, “Auditoria_segguimiento” y “Cedula_observaciones”.
- **Detalle de observaciones** que genera una consulta a la tabla “Cedula_observaciones”.
- **Resumen de observaciones** de auditoría que genera una consulta a las tablas “Auditoria”, “Auditoria_segguimiento” y “Cedula_observaciones”.
- **Total de observaciones** que genera una consulta a la tabla “Cedula_observaciones”.

La segunda sección implementa filtros para los reportes **Detalle de observaciones**, **Resumen de observaciones** y **Total de observaciones**. Estos filtros se aplican sobre los campos de la tabla “Cédula observaciones”.

La tercera sección implementa filtros sobre los campos de la tabla “Auditoria” para todos los reportes que genera el módulo. El criterio de filtrado **Nueva orden** o **Auditoría**

seguimiento sólo se implementó para los reportes **Resúmenes de orden de auditoría y Resumen de observaciones** de auditoría.

Por último la sección de botones de control posee cuatro funciones básicas: generar reportes en pantalla (preview que sirve al usuario para verificar los datos obtenidos al filtrar antes de imprimir el reporte), generar reportes a impresora, limpiar criterios de filtrado para reiniciar una consulta y salir del módulo. Cabe resaltar el empleo de *Quick Report* para la generación de los reportes en pantalla. *Quick Report* de *Delphi* es el equivalente a *Crystal Reports* de *Visual Basic*, y permite generar de manera sencilla formatos de reporte que reciben consultas de SQL y les dan presentación. En este módulo no se empleo OLE por que no fue necesario interactuar con Excel o Word.

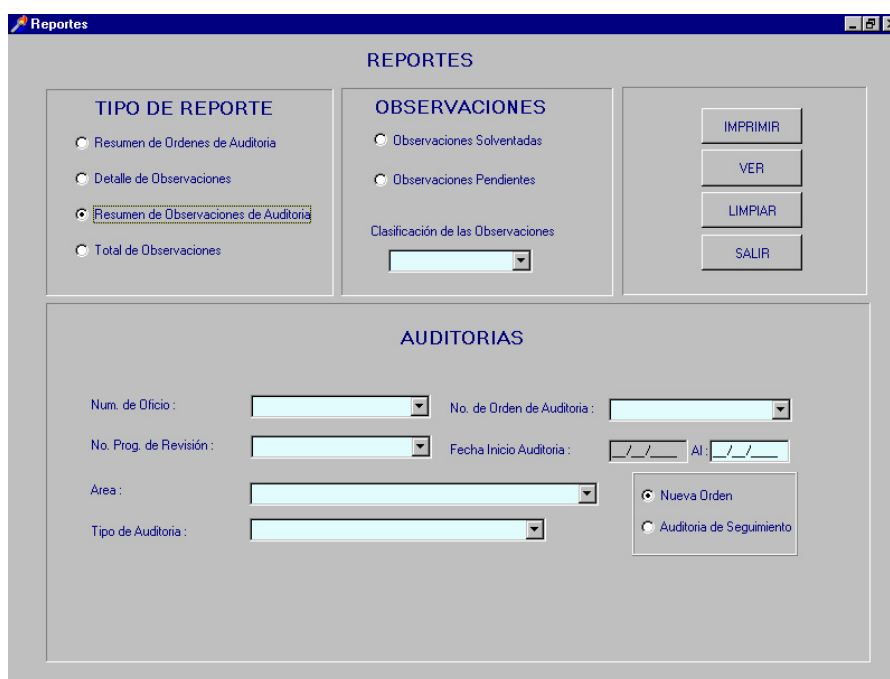


Figura 5.11 Pantalla Reportes

5.2.2 Supervisión

Es común que los programadores tiendan a modificar los lineamientos del desarrollo. A menudo, esto sucede cuando el equipo de trabajo tiene poca experiencia trabajando junto o cuando alguno de sus integrantes es nuevo. La supervisión del desarrollo de los módulos es muy importante pues de ella depende el correcto funcionamiento de los módulos conforme a los diseños y lineamientos establecidos previamente.

La supervisión que se llevó a cabo durante el período de desarrollo de “SAPA”, consistió en una serie de reuniones semanales en las que se evaluó: el avance del la codificación de cada

módulo, el apego a los estándares de programación definidos, la presentación de las pantallas desarrolladas (revisión de cosméticos) y revisión de la funcionalidad conforme a los diseños.

El responsable de la supervisión es el líder de proyecto por medio de los diseños y del código de la aplicación. Los programadores tienen la obligación de realizar y notificar de cualquier modificación al diseño al líder de proyecto durante el desarrollo de estas juntas. Se deben elaborar minutas en las cuales se registren los asuntos tratados en estas reuniones.

5.3 Documentación

Consiste en la creación de dos manuales que sirven al usuario y a los responsables del mantenimiento del sistema como base de su trabajo.

El primero de los documentos importantes es el “Manual de Usuario” y lo complementa el “Manual Técnico”.

5.3.1 Manual de Usuario

El manual de usuario es un documento en el que se encuentra definida la operación general del sistema. Este documento es de gran utilidad para el usuario, pues con él podrá resolver todas sus dudas de funcionamiento, validación o búsqueda de algún dato, errores que produce el sistema, lo que evitará la necesidad de consulta con el desarrollador del sistema. El manual de usuario también es una herramienta útil durante el proceso de capacitación de los usuarios.

Los diferentes módulos del manual de usuario fueron documentados por cada uno de los programadores responsables del desarrollo (nadie mejor para describir el funcionamiento de un módulo que el que lo construyó). El inconveniente de trabajar de esta manera es que cada uno de los programadores tiene un estilo de documentación que hay que unificar por medio de un estándar de documentación.

El manual de usuario debe de contener lo siguiente:

- *Definición del sistema y principales características del mismo.*- Se deberá explicar el propósito para el cual fue creado el sistema.
- *Estándares a utilizar en el manual.*- Se describen los estándares que se emplean en el manual.
- *Seguridad.*- Breve explicación del manejo de la seguridad del sistema.
- *Acceso al sistema.*- Breve explicación de cómo iniciar la aplicación desde el sistema operativo en donde está instalada (en este caso Windows 9x o NT) y una descripción del uso de la pantalla de acceso.
- *Procedimientos generales.*- Explicar los procedimientos que se repiten constantemente durante la ejecución de la aplicación, para evitar describirlos en cada tema del manual. Un ejemplo es: el uso del teclado y los botones generales. La descripción de uso de los

botones se llevará a cabo en una tabla en la que se muestra como primer columna la imagen del botón y en la segunda su descripción. A continuación se muestra un ejemplo:


Botón	Descripción
	Una vez capturados todos los datos del documento, se oprime este botón para generar el reporte correspondiente.

Tabla 5.2 Procedimientos generales

En esta sección también se incluye la ventana principal del sistema y su descripción.

- *Funcionamiento del sistema.*- Son los procedimientos para usar el sistema. La organización del documento debe depender de las características del software a ser documentado, puede ser:
 - Basándose en cómo el usuario trabaja con el sistema.
 - Por los menús con que cuenta el sistema.

A continuación se muestran parte de los estándares para la redacción del manual de usuario:

Sección	Estilos	Estilo dentro de Word	Tamaño (puntos)	Con margen
Título del manual (Primera página del manual)	Title	Arial Bold	36	Alineado a la derecha
Subtítulo del manual	Supertitle	Arial Bold	16	Alineado a la derecha
Título de los capítulos	Heading 1	Arial Bold	36	no
Títulos 1	Heading 2	Arial Bold	20	Izquierdo
Títulos 2	Heading 3	Arial Bold	16	Si(4 cm)
Títulos 3	Heading 4	Arial Bold	12	Si(4 cm)
Títulos 4	Heading 5	Arial Bold	10	Si(4 cm)
Textos dentro del manual	Body Text	Times New Roman	12	Si(4 cm)

Tabla 5.3 Estándares de redacción

Para ayudar a localizar e interpretar la información contenida dentro del manual de una manera más sencilla, se utilizarán las siguientes convenciones:

Texto	Característica	Estilo	Tamaño
Teclas rápidas Ej: ALT+CTRL	Minúsculas tipo SMALL CAPS	Times New Roman	12
Menús Ej: Clases	Primera letra con mayúscula	Times New Roman Bold	12
Directorios y nombres de	Mayúsculas	Times New Roman	12

archivos Ej: C:\SAPA\ORDEN.XLS			
Nombres de Botones Ej: Cambios	Primera letra con mayúscula	Times New Roman Bold	12
Nombres de los campos Ej: Dirección	Primera letra con mayúscula	Times New Roman Bold	12
Referencias a temas dentro del manual Ej: “Orden de Auditoría”	Debe estar entre comillas y la primera letra de cada palabra con mayúscula.	Times New Roman Bold	12

Tabla 5.4 Convenciones de redacción

1

Para el manejo de cualquier tabla en el manual se tomarán en cuenta los siguientes estándares:

	Estilo o fuente	Negrita	Tamaño	Sombreado
Estilo de tabla	Elegante			
Títulos	Arial	Bold	10	10 %
Textos	Times New Roman	Normal	12	Ninguno

Tabla 5.5 Estándares de tablas

Con respecto al encabezado y pie de página se tiene:

	Características	Estilo en Word	Tamaño(puntos)
Encabezado	Borde Superior	Arial Bold	9
Pie	Borde Inferior	Arial Bold	9

Tabla 5.6 Estándares de encabezado y pie de página

Los elementos que contendrán son los siguientes:

Encabezado par:

Número de Capítulo – Número de Página

Nombre del Capítulo

Encabezado impar:

Nombre del Capítulo

Número de Capítulo – Número de Página

Pie de página par:

Logo de la empresa

Nombre de la aplicación

Pie de página impar:

Nombre de la aplicación

Logo de la empresa

El desarrollo de cada tema en el manual debe seguir el siguiente formato:

- Título del tema u opción del menú
- Datos generales
 - Descripción general del tema.
 - Descripción de cómo acceder la opción.
 - Imagen de la Ventana (al 80% de su tamaño original).
- Funcionamiento
 - Explicar al usuario cómo debe de trabajar en la pantalla.

En la portada del manual deberá ir el logotipo de la empresa para la cual se realizará la documentación, el nombre del manual a realizar y el Nombre del Sistema.

En la portada de cada uno de los capítulos se deberán incluir los siguientes elementos:

- Número del capítulo
- Título del capítulo
- Imagen que se relacione con el contenido del capítulo

5.3.2 Manual Técnico

El manual técnico es un documento que sirve para documentar las diferentes partes que conforman un sistema y que no son visibles al usuario. Esta documentación está principalmente dirigida al equipo de trabajo que dará mantenimiento al sistema después de su liberación. El manual técnico deberá constar de lo siguiente:

- *Un índice o contenido.*
- *Un tema en el cual se explicarán las principales características de la aplicación desarrollada (se debe explicar el propósito para el cual fue creada la aplicación).*
- *Instalación del sistema.-* Deben explicarse los requerimientos de hardware y software del sistema, explicar cómo instalar el sistema y mostrar cómo se lleva a cabo su configuración.
- *Seguridad.-* Debe explicarse el esquema de seguridad del sistema.
- *Formas del sistema.-* Se incluirá una descripción de todas las formas del sistema junto con los stored procedures y las tablas que emplean.
- *Funciones generales del sistema.-* Consta de una lista de todas las funciones generales del sistema (públicas o privadas), con una pequeña descripción y los parámetros que utiliza.
- *Stored procedures de la aplicación.-* Consta de una lista de todos los SP(Stored procedures) utilizados en el sistema conteniendo descripción, función(es) que lo(s) invoca(n), parámetros dentro del stored y su descripción, tablas que accesa y los stored procedures con los que se relaciona.
- *Diccionario de datos y DER.-* En este tema se explicarán cada una de las tablas que integran al sistema, así como la descripción de cada uno de sus campos. De cada campo se debe explicar su tipo de dato, longitud y descripción. El diccionario de datos y el DER se crean desde la fase de análisis, por lo que prácticamente solo hay que integrarlos al manual técnico.

- *Fallas y soluciones.*- Se describen los problemas que pueden presentarse durante la ejecución del sistema y sus posibles soluciones.

El “**Manual Técnico**” sigue los mismos estándares de tipo de letra, encabezados y pies de página, tablas, etc., descritos para el manual de usuario.

5.4 Pruebas

5.4.1 Pruebas de Validación

Este tipo de prueba consiste en verificar que los datos de entrada sean válidos por sí mismos. Para asegurar lo anterior se pueden considerar las siguientes pruebas:

- *Prueba de datos faltantes.*- Verifica que no falten datos. En ciertas situaciones los datos deben estar presentes. Por ejemplo, al dar de alta una orden de auditoría, es requisito asignarle un auditor responsable. Este criterio se emplea para permitir a los auditores que no sean administradores del sistema consultar únicamente las auditorías de las cuales son responsables. El analista de sistemas debe relacionarse con los usuarios para determinar qué datos son esenciales y percatarse si hay casos excepcionales que se acepten como datos válidos, a pesar de encontrarse incompletos.
- *Prueba de la longitud del campo.*- Verifica que la entrada cuente con la longitud del campo correcta. Por ejemplo si no se proporciona la longitud adecuada de la fecha de elaboración en el “**PACA**” no se puede salir del campo.
- *Prueba del tipo o composición.*- Verifica en los datos proporcionados sean congruentes con el tipo de dato del campo en la base donde serán almacenados. Es decir, si en una pantalla se captura un tipo entero, el campo donde se almacenará en la base de datos debe de ser entero, o si es de otro tipo de mayor precisión (float o double) o texto, debe existir una conversión que evite errores. En este caso es importante conocer las conversiones automáticas que llevan acabo algunos lenguajes o las bases de datos (normalmente esto se da de numérico a cadena).
- *Prueba de rango o racionalidad.*- Verifica que los valores introducidos en un sistema caigan dentro de un rango aceptable o dentro de parámetros establecidos. Por ejemplo, si un usuario trata de introducir la fecha 01/15/2003, el sistema manda un mensaje de error por que no permite fechas con un número de mes mayor a 12 (el formato de fechas es dd/mm/aaaa)
- *Pruebas para valores no válidos.*- Funciona si solo conforman una pequeña fracción. Esta prueba es de poco valor en el caso de que se cuente con un conjunto de datos restringidos o predecibles. La prueba es útil para verificar aquellas respuestas donde los datos se dividen en un número limitado de clases.
- *Prueba de comparación de datos ya almacenados.*- Compara contra los datos la base de datos. Esta validación se lleva normalmente a cabo contra los datos que pertenecen a los catálogos del sistema, pues es ahí donde se almacena la información constante del sistema.

Durante las pruebas de validación se decidió capturar los datos históricos que requería el sistema para su puesta en operación. Los datos de prueba utilizados fueron:

- PACA de 2001,2002 y 2003.
- Documentos completos de todas las auditorías nuevas y de seguimiento con observaciones pendientes.
- El total de las observaciones pertenecientes al inventario de observaciones vigente.
- Catálogo de personal de la institución gubernamental.
- Catálogo de tipos de auditorías.
- Catálogo de áreas de la institución gubernamental.

Al llevar acabo la captura de los datos históricos fue necesario hacer una revisión final de la congruencia de la información contenida en la base de datos contra los documentos proporcionados por el área de Contraloría.

5.4.2 Pruebas de Unidad

Estas pruebas consistieron en evaluar cada uno de los módulos de forma independiente, recorriendo una a una las pantallas que los conforman. Durante esta fase de prueba se realiza un calendario de pruebas en el cual se incluye la participación del usuario con el fin de obtener la opinión del funcionamiento de c/u de los módulos. En esta fase se explicó al usuario la funcionalidad completa de cada módulo. En algunos partes de la operación de los módulos el usuario no entendía el por qué de su funcionamiento, pero al explicárselo detalladamente, en la mayoría de los casos quedó satisfecho. Para los casos en que no se llegó a un acuerdo sobre la operación de partes de los módulos, se tomó nota para corregirlos y presentarlos corregidos en la fase de pruebas integrales.

5.4.3 Pruebas de Integración o Integrales

Esta es la fase final de pruebas y es en la que se evalúa el funcionamiento integral del sistema. Para llevar a cabo esta prueba se definieron objetivos a alcanzar:

- Capturar de manera completa dos revisiones del PACA del año 2003.
- Capturar toda la información correspondiente a la nueva auditoría 01/2003 y generar sus respectivos documentos:
 - Orden de auditoría.
 - Acta de inicio.
 - Carta planeación.
 - Cronograma de actividades.
 - Un Marco conceptual.
 - Cédulas de observaciones correspondientes.
 - Informe de auditoría y cédula única de auditoría.
 - Evaluación de auditoría.
- Probar la correcta integración de las observaciones de la nueva auditoría al inventario, por medio de la opción procede o no procede, así como su correcta reenumeración.
- Capturar toda la información correspondiente a una auditoría de seguimiento 02/2003(prueba con datos falsos) y generar sus respectivos documentos:
 - Orden de auditoría.
 - Acta de inicio.

- Carta planeación.
- Cronograma de actividades.
- Un Marco conceptual.
- Cédulas de seguimientos correspondientes.
- Informe de auditoría y cédula única de auditoría.
- Evaluación de auditoría.
- Emplear en la elaboración de la auditoría de seguimiento el inventario de observaciones. Llevar a cabo filtrado de observaciones.
- Modificar el avance de solventación de las observaciones a las que se les dio seguimiento para mostrar su comportamiento en el inventario.

CAPÍTULO 6



*Implantación,
Evaluación y
Mantenimiento*

6.1 Implantación

En esta etapa se ayuda a implantar el sistema, esto incluye la capacitación de los usuarios, la verificación de la correcta instalación y su puesta en operación.

6.1.1 Verificación de Correcta Instalación

El primer punto de la implantación de un sistema es la instalación del mismo en el lugar donde va a operar. Parece que esto se lleva a cabo tan sólo en una fase, pero en realidad no es así. Primeramente, se creó un archivo instalador (*Setup*), que incluyera todos los archivos propios del sistema, utilerías (dll's, etc.), formatos o plantillas de documentos oficiales, *DBEngine* de *Delphi* (Manejador de conexiones a bases de datos de *Delphi*), e íconos que sirven para identificar el software. En segundo lugar se hicieron pruebas de instalación del software en PC's ajenas al área de contraloría para determinar posibles fallas de instalación y verificar el desempeño del sistema. Al mismo tiempo se verificó la operación del sistema desde diferentes PC's para eliminar errores de concurrencia y errores de conexión a la base de datos del servidor. Finalmente, se acordó con el área de contraloría un calendario de instalación del software en todas y cada una de las PC's de los usuarios del área. Al término de la instalación del software se llevó a cabo una prueba integral de funcionamiento del sistema.

Cabe señalar que paralelamente a la instalación del software, se capacitó al jefe de departamento en la instalación del software y se le entregó una copia del archivo de instalación para futuras instalaciones.

6.1.2 Capacitación de los Usuarios

Para la correcta operación de un sistema siempre será indispensable capacitar al usuario en el uso del mismo. Capacitar es un proceso educativo en el que se vinculan de manera sencilla y práctica los conocimientos del análisis de un sistema con el usuario.

La capacitación se impartió a todos los usuarios que interactúan con el sistema: capturistas de observaciones, jefes de unidad (contadores que llevan a cabo las auditorías), jefe de departamento (supervisor), director de área (responsable de la toma de decisiones sobre las observaciones de las auditorías) y director general (responsable del área).

Se consideró que las personas adecuadas para impartición de la capacitación serían el analista de sistemas y los usuarios con los que se definieron los requerimientos del sistema (en este caso sólo uno). La experiencia del analista de sistemas en cursos de capacitación a usuarios fue un punto a considerar para la estrategia de impartir un curso de tres horas a toda el área. Otra estrategia de capacitación fue la consultoría directa del usuario con el que se definieron requerimientos del sistema por parte de todos los usuarios del área.

En la capacitación por medio de curso se empleó un método que incluyó una demostración práctica del sistema, proyectada directamente a los usuarios. Lo ideal hubiera sido contar con una sala de cómputo con pantalla para que los usuarios pudieran practicar al mismo tiempo que observaban la demostración de la aplicación, y que el curso durase al menos tres horas al día durante una semana, pero el tiempo no fue suficiente, por lo que sé, el método de enseñanza por medio de un manual de usuario fue el apoyo que complementó la capacitación.

El material para el adiestramiento constó de proyector de pantalla de computadora, computadora con sistema instalado, conexión a red interna, mesas para el adiestrador y alumnos, así como copias del manual de usuario. Además, cada usuario contó con un cuaderno de anotaciones.

6.1.3 Puesta en Operación

Para la puesta en operación del sistema fueron necesarios los requerimientos de software y hardware descritos en la sección de **análisis de requerimientos** y acordar con el área un calendario para la instalación.

El siguiente paso consistió en la definición del tipo de conversión a aplicar entre la operación del sistema vigente y la operación con el nuevo sistema informático. Existen cinco tipos de conversiones comunes:

- *Reemplazo total.*- Consiste en el inmediato reemplazo del sistema anterior con el nuevo.
- *Conversión en paralelo.*- Es cuando los dos sistemas (actual y nuevo), operan de manera conjunta hasta que el nuevo logra cubrir las responsabilidades del sistema a reemplazar; en ese momento deja de funcionar el sistema anterior.
- *Conversión gradual.*- Consiste en el incremento gradual del empleo del nuevo sistema reemplazando una a una las responsabilidades del sistema viejo.
- *Conversión por prototipos modulares.*- Se refiere al reemplazo gradual de módulos del viejo sistema por módulos del nuevo sistema.
- *Conversión distribuida.*- Es cuando se contempla la instalación de un sistema en múltiples máquinas y se empieza por instalar sólo en algunas y probar hasta que esa instalación es correcta.

En el caso de “SAPA” se empleó una combinación entre el reemplazo total y la conversión en paralelo, pues fue necesario que, por un período de tiempo, los auditores responsables de la generación de las auditorías lo hicieran con el método nuevo (por medio del sistema), pero respaldados, en caso de falla, por el sistema tradicional; es decir, si el sistema fallaba a la hora de generar documentos, almacenar información, ejecutar validaciones, consultar algún dato capturado o ejecutar algún proceso, el usuario siempre tuvo la opción de generar los documentos por medio del procesador de texto y de consultar la información histórica por medio de la consulta de los documentos de la auditoría (método tradicional).

6.2 Evaluación

La evaluación de un sistema de información es un proceso que tiene como fin el perfeccionamiento del mismo y también puede servir como un método de seguimiento a la implantación. La evaluación final es la que recoge todas las impresiones de la operación del sistema. La evaluación del sistema es el hecho de que el usuario realmente llegue a usar el sistema, satisfaciendo sus necesidades de información.

6.2.1 Técnicas de Evaluación

Existen múltiples técnicas de evaluación, por ejemplo, los análisis de costo beneficio que se pueden aplicar pero cuentan con la desventaja de no poder aplicar siempre las comparaciones que requiere (en un sistema no siempre es posible la comparación entre un antes y un después porque un sistema genera nuevas maneras de considerar los procesos o la información). La evaluación de decisiones es otra técnica de evaluación de sistemas pero cuenta con la problemática de no poder cuantificar todos los valores involucrados en el desarrollo de un sistema.

El enfoque de utilerías del sistema es la técnica que se empleó para la evaluación de “SAPA” y consiste en la abstracción de módulos, pantallas, reportes o cualquier parte de un sistema como una utilería evaluable. Las utilerías pueden ser de posesión, forma, lugar, tiempo, actualización y objetivo. La utilería de posesión se refiere a si la persona adecuada tiene la información adecuada; la utilería de forma se refiere a que la información tenga el correcto formato; la utilería de lugar indica si la información se está distribuyendo en el lugar indicado; la de tiempo se refiere a si la información está llegando en el momento adecuado; la de actualización indica si la información está siendo actualizada de manera correcta, y, por último, la utilería de objetivo es aquella que nos define si la información cumple con los objetivos del tomador de decisiones.

6.2.2 Evaluación del Sistema

Si un sistema posee las seis utilerías descritas anteriormente puede considerarse como un sistema exitoso. Al llevar a cabo la evaluación de “SAPA” por medio de utilerías se eligió descomponer la evaluación de manera modular para tener una mejor revisión del sistema. A continuación se muestra la tabla de evaluación de “SAPA”:

Módulo	Utilería					
	Forma	Tiempo	Lugar	Posesión	Actualización	Objetivos
Carta Planeación	Buena.- Se ajusta a los formatos oficiales	Buena.- Permite la captura y generación de todos los documentos necesarios por año	Buena.- La información está disponible para impresión en todo el área filtrado por auditor	Buena.- La información está disponible para consulta en todo el área filtrado por auditor (admin consulta todo)	Buena.- La información está disponible para modificación en todo el área filtrado por auditor (admin modifica todo)	Buena.- Creación y almacenamiento automático de documento oficial (consulta de datos históricos automático)
Cronograma de Actividades	Buena.- Se ajusta a los formatos oficiales	Buena.- Permite la generación de todos los documentos necesarios por año	Pobre.- La información no está disponible para impresión por que no se almacena en la base	Pobre.- La información no está disponible para consulta por que no se almacena en la base	Pobre.- La información no está disponible para modificación por que no se almacena en la base	Regular.- Creación automática de documento oficial (consulta de datos históricos automático)
Marco Conceptual	Buena.- Se ajusta a los formatos oficiales	Buena.- Permite la captura y generación de todos los documentos necesarios por año	Buena.- La información está disponible para impresión en todo el área filtrado por auditor	Buena.- La información está disponible para consulta en todo el área filtrado por auditor (admin consulta todo)	Buena.- La información está disponible para modificación en todo el área filtrado por auditor (admin modifica todo)	Buena.- Creación y almacenamiento automático de documento oficial (consulta de datos históricos automático)
Cédula de Observaciones	Buena.- Se ajusta a los formatos oficiales	Buena.- Permite la captura y generación de todos los documentos necesarios por año	Buena.- La información está disponible para impresión en todo el área filtrado por auditor	Buena.- La información está disponible para consulta en todo el área filtrado por auditor (admin consulta todo)	Buena.- La información está disponible para modificación en todo el área filtrado por auditor (admin modifica todo)	Buena.- Creación y almacenamiento automático de documento oficial (consulta de datos históricos automático)
Cédula de Seguimiento	Buena.- Se ajusta a los formatos oficiales	Buena.- Permite la captura y generación de todos los documentos necesarios por año	Buena.- La información está disponible para impresión en todo el área filtrado por auditor	Buena.- La información está disponible para consulta en todo el área filtrado por auditor (admin consulta todo)	Buena.- La información está disponible para modificación en todo el área filtrado por auditor (admin modifica todo)	Buena.- Creación y almacenamiento automático de documento oficial (consulta de datos históricos automático)
Informe de Auditoría	Buena.- Se ajusta a los formatos oficiales	Buena.- Permite la captura y generación de todos los documentos necesarios por año	Buena.- La información está disponible para impresión en todo el área filtrado por auditor	Buena.- La información está disponible para consulta en todo el área filtrado por auditor (admin consulta todo)	Buena.- La información está disponible para modificación en todo el área filtrado por auditor (admin modifica todo)	Buena.- Creación y almacenamiento automático de documento oficial (consulta de datos históricos automático)
Evaluación de auditoría	Buena.- Se ajusta a los formatos oficiales	Buena.- Permite generación de todos los documentos necesarios por año	Pobre.- La información no está disponible para impresión por que no se almacena en la base	Pobre.- La información no está disponible para consulta por que no se almacena en la base	Pobre.- La información no está disponible para modificación por que no se almacena en la base	Regular.- Creación automática de documento oficial (consulta de datos históricos automático)
Catálogos		Buena.- Permite la captura de información y disponibilidad de la misma en cualquier momento	Buena.- La información está disponible para impresión en toda el área	Buena.- La información está disponible para consulta en toda el área	Buena.- La información está disponible para modificación en toda el área	Buena.- Almacenamiento histórico de datos generales
Reportes	Buena.- Se ajusta a los formatos oficiales	Buena.- Permite la captura y generación de todos los documentos necesarios por año, trimestre, mensual o diario	Buena.- La información está disponible para impresión en toda el área	Regular.- La información está disponible para consulta en toda el área y debió ser restringida	Buena.- La información está disponible para modificación en toda el área	Buena.- Generación de información concentrada para informes anuales, semestrales, trimestrales o mensuales.

Tabla 6.1 Evaluación de utilerías

Módulo	Utilería					
	Forma	Tiempo	Lugar	Posesión	Actualización	Objetivos
PACA	Buena.- Se ajusta los formatos oficiales	Buena.- Permite la captura una vez año	Buena.- La información está disponible para impresión en toda el área	Buena.- La información está disponible para consulta en toda el área	Regular.- La información está disponible para modificación en toda el área y sólo debería ser modificada por el jefe de depto. y sus superiores	Buena.- Información disponible para el proceso de auditoría
Orden de Auditoría	Buena.- Se ajusta los formatos oficiales	Buena.- Permite la captura y generación de todos los documentos necesarios por año	Buena.- la información está disponible para impresión en toda el área filtrada por auditor	Buena.- La información está disponible para consulta en todo el área filtrado por auditor (admin. consulta todo)	Buena.- La información está disponible para modificación en todo el área filtrado por auditor (admin modifica todo)	Buena.- Creación y almacenamiento automático de documento oficial
Consulta de inventario	Buena.- Emplea los datos representativos	Buena.- Permite la consulta en el momento que se desee.	Buena.- La información está disponible para impresión en toda el área	Buena.- La información está disponible para consulta en toda el área	Buena.- Consulta todas las observaciones generadas en el área	Buena.- Inventario informático de observaciones
Acta de Inicio	Buena.- Se ajusta a los formatos oficiales	Buena.- Permite la captura y generación de todos los documentos necesarios por año	Buena.- La información está disponible para impresión en todo el área filtrado por auditor	Buena.- La información está disponible para consulta en todo el área filtrado por auditor (admin consulta todo)	Buena.- La información está disponible para modificación en todo el área filtrado por auditor (admin modifica todo)	Buena.- Creación y almacenamiento automático de documento oficial (consulta de datos históricos automático)

Tabla 6.2 Evaluación de utilerías (continuación)

La conclusión de la evaluación de “SAPA” es positiva pero será necesario aplicar cambios en la fase de mantenimiento perfectivo para mejorar aquellas utilerías que son regulares o pobres.

6.3 Mantenimiento

El mantenimiento del sistema empieza en esta etapa y, después, se realizará de forma rutinaria a lo largo de toda la vida del sistema. Mientras mejor sea el análisis, diseño y desarrollo de un sistema, el tiempo y costo de mantenimiento será menor.

6.3.1 Mantenimiento Preventivo

Consiste en la detección, por medio de la evaluación y pruebas, de los errores que pueden llegar a ser potenciales fuentes de inconsistencias o fallas dentro del sistema. Este tipo de mantenimiento es el que suele darse durante la fase de conversión y al principio de la fase de mantenimiento y es importante desarrollarlo rápidamente para evitar contingencias al inicio de la operación del sistema. En “SAPA”, los reportes fueron sometidos a mantenimiento perfectivo.

6.3.2 Mantenimiento Correctivo

Durante la operación inicial del sistema suelen presentarse errores que requieren de corrección para su correcta operación. El mantenimiento correctivo se enfoca en la solución de errores; algunos pueden ser detalles que sólo afectan alguna parte del sistema, sin embargo, puede haber otros que afecten el funcionamiento general y son estos errores los que tienen prioridad de atención. Otro factor que influye en la prioridad de atención de errores es el conocer cuáles de ellos son los que afectan las partes críticas del sistema (sistema operativo conexión a redes, conexión a bases de datos, etc.), o la integridad de la información que se almacena. El mantenimiento correctivo que se aplicó a “SAPA” permitió hacer más eficiente el uso del inventario de observaciones pues en el manejo de los porcentajes de solventación de las observaciones no se integraban de manera adecuada. Otro error corregido, de hecho con más alta prioridad de corrección, fue el correcto funcionamiento del módulo PACA (los botones de navegación presentaron funcionamientos sin validaciones de Resultset vacío y fallaban).

6.3.3 Mantenimiento Perfectivo

Este tipo de mantenimiento se aplica para mejorar el funcionamiento de un sistema. Normalmente implica adicionar funcionalidad propuesta por los usuarios. Es común que muchas de las ideas que tienen los usuarios para el funcionamiento del sistema se queden en el tintero pues el tiempo de desarrollo de un proyecto es limitado, por lo anterior, una buena fuente de posibles mejoras para el sistema es aquellas cosas que el usuario normalmente propone cuando ya se ha concluido la fase de análisis y diseño. Los usuarios también pueden proponer ideas de mejora que surjan durante su cotidiana interacción con el sistema. En “SAPA”, algunas de las posibles mejoras incluyen: la implementación de un proceso de lectura de un archivo de texto que contenga el PACA, la optimización del proceso de reenumeración de las observaciones (eliminando una tabla que parece no ser necesaria), la integración y eliminación de campos y controles innecesarios en el módulo “PACA” y la optimización de el uso de Resultset.



Conclusiones

CONCLUSIONES

Originalmente el proceso de auditoría se llevaba a cabo empleando una serie de documentos que duplicaban datos y que no tenían una uniformidad en la captura, provocando inconsistencias en la información. Este proceso también generaba pérdida de horas hombre de trabajo a causa de tres factores: la captura redundante de información contenida en otros documentos de la misma auditoría, la búsqueda en archivo de los documentos de auditoría y la supervisión manual de documentos de auditoría por parte del supervisor de auditoría.

El propósito SAPA fue resolver todos los problemas planteados por medio del desarrollo de un sistema computacional sencillo de emplear y que permitiera el manejo y control de la información de manera eficiente aplicando los conocimientos adquiridos durante el tiempo de estudio de la carrera de Ingeniero en Computación.

El empleo de SAPA se traduce en un considerable ahorro de tiempo en el desarrollo del proceso de auditoría por que permite la generación y consulta de manera automática de las auditorías a través de pantallas que emulan cada uno de los documentos empleados en el proceso (orden de auditoría, acta de inicio), así como su generación en Word o Excel.

SAPA también permite el ahorro de tiempo en la consulta de una o del total de auditorías y de sus observaciones, proceso comunmente llevado a cabo por el supervisor de auditoría y los directores del área.

El seguimiento de las observaciones generadas en el proceso de auditoría, así como su almacenamiento se refleja en toda la auditoría de manera automática, permitiendo un mejor control de las auditorías de seguimiento y de las observaciones.

El control de la consistencia de la información es más eficiente debido a la eliminación de redundancias en la captura.

El uso de SAPA plantea una substitución gradual del almacenamiento en documentos de papel, por información electrónica que permita su mejor manipulación informática y consulta más dinámica que en un archivo tradicional, además con el tiempo, esta información podría ser compartida con otros organos de control interno de otras dependencias de gobierno o con la propia Secretaría Gubernamental que coordina los órganos de control interno de las instituciones de gobierno.

En el aspecto de implementación de la aplicación, el empleo de una metodología de desarrollo de sistemas (clásica), permitió trabajar de manera más organizada y eficiente, traduciéndose esto en la reducción del tiempo de programación, debido a un mayor enfoque y dedicación a la etapa de análisis y diseño, y aún más, en una esperada reducción en el tiempo que tenga que darse al mantenimiento correctivo y perfectivo del producto una vez liberado.

El empleo de la metodología Clásica auxilió en la fácil implantación de un modo metodológico de trabajo entre jóvenes recién egresados de estudios de licenciatura y

proporciona las bases para tomar metodologías de más vanguardia como RUP (Rational Unified Process) o (Metodología orientada a objetos de SUN).

La modularidad en la que se basa el desarrollo de SAPA, permite la clara identificación de cada una de las etapas del proceso de auditoría y por lo tanto también permite su adaptabilidad a los posibles cambios y/o adiciones que sufra el proceso.

El haber empleado un lenguaje de desarrollo como lo es Object Pascal (lenguaje propio de DELPHI), que es orientado a objetos (aunque empleado en este proyecto de manera estructurada) permite que en un futuro la aplicación pueda ser modelada y adaptada a una metodología mas reciente como USDP.

Como estudiante, la conclusión principal gira en torno al beneficio que proporcionó el llevar a la práctica los conocimientos teóricos adquiridos en la carrera, en especial aquellos que tienen relación con las materias de ingeniería de programación, bases de datos, programación estructurada, organización de computadoras y redes principalmente.

Como conclusión general se puede tener que SAPA es un sistema que permite un mejor aprovechamiento, organización y control de la información relacionada al proceso de auditoría, teniendo como consecuencia directa un incremento en la eficiencia y velocidad con la que pueden desarrollarse, tanto el proceso de auditoría ejecutado por un auditor, como la supervisión del total de auditorías generadas en un área de contraloría interna de cualquier institución gubernamental.



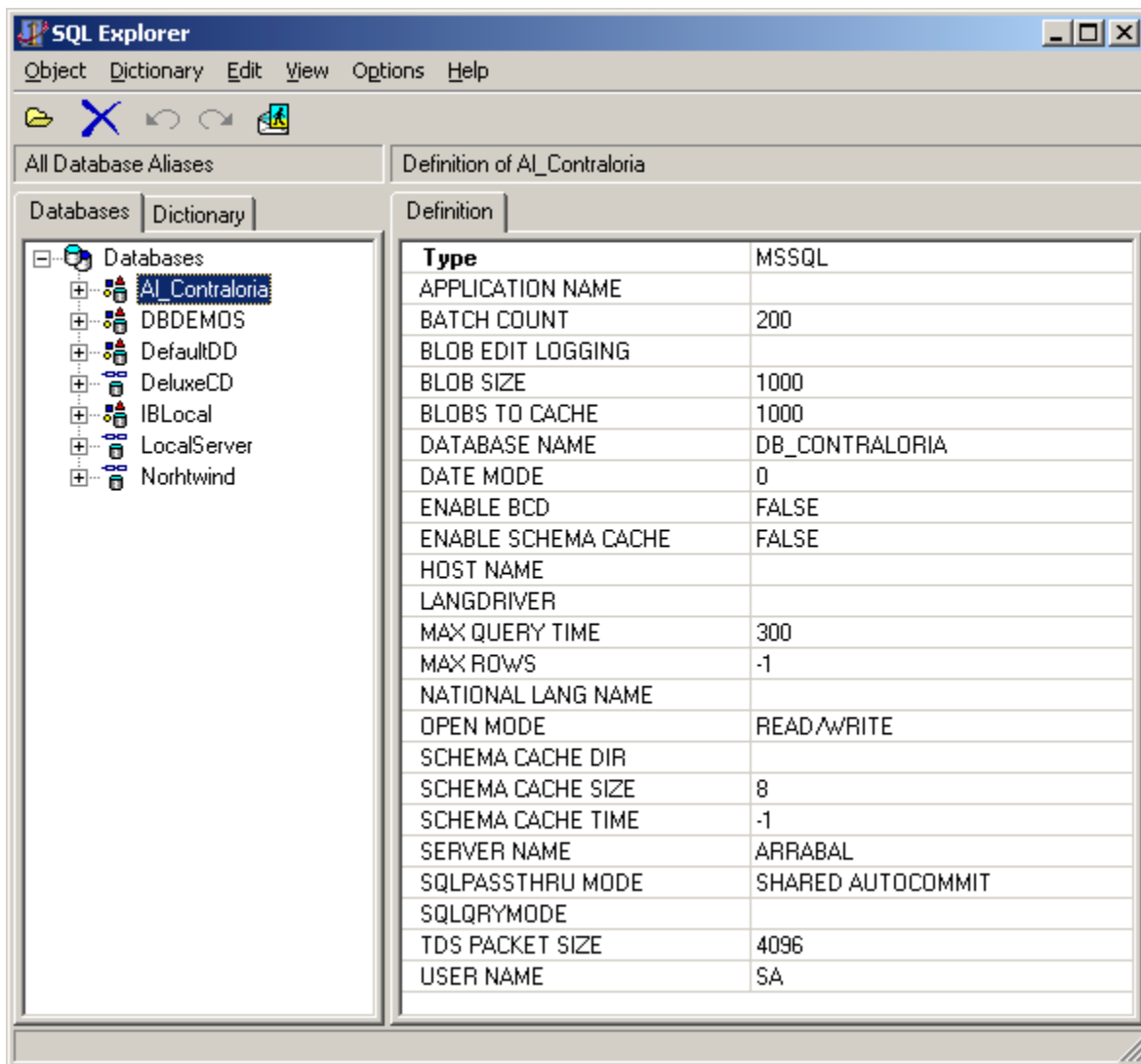
Apéndices

APÉNDICE A

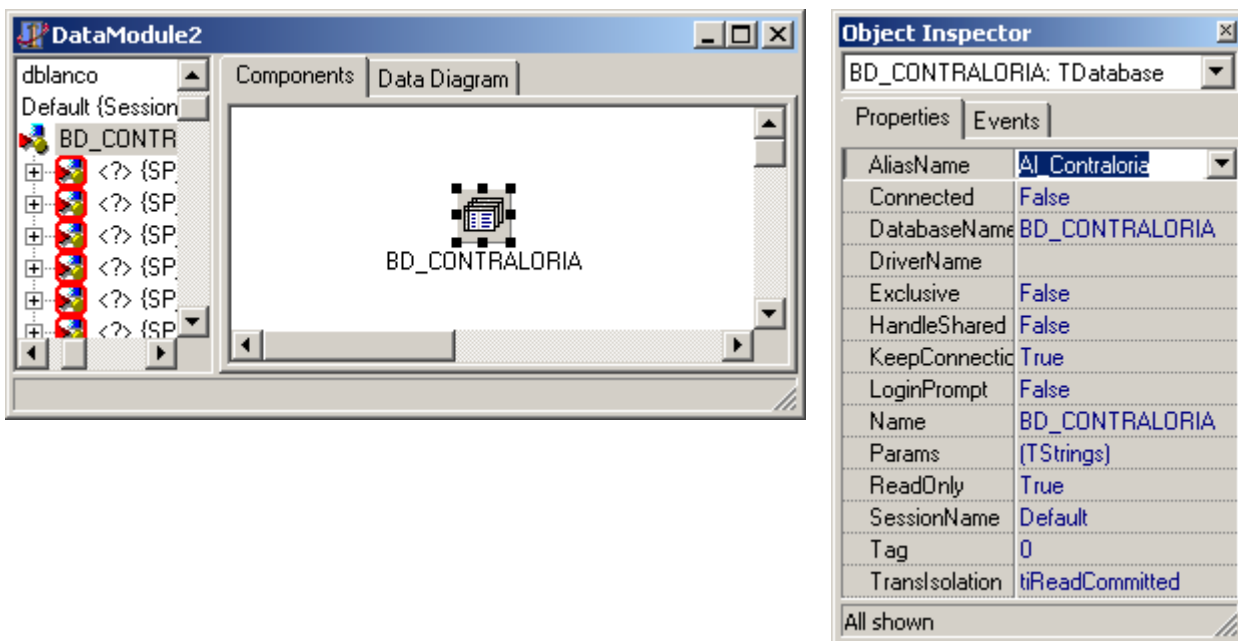
Conexión y Acceso a Base de Datos

Conexión

“SAPA” se conecta a la base de datos por medio del *Delphi BDE (Borland Database Engine)*. El *BDE* viene a cumplir la misma función de *ODBC* de Windows. Al instalar el sistema en un cliente, se instala el *BDE Administrator* (herramienta de administración de *BDE*), los controladores (drivers) de conexión al DBMS elegido (*Microsoft SQL Server*) y se configura un alias de conexión a la base de datos (homólogo a un *DSN* para *ODBC*). A continuación se muestra una imagen del *BDE Administrator* en donde se puede observar los datos de configuración del alias “*Al_Contraloria*” que se emplea para conectar “*SAPA*”:

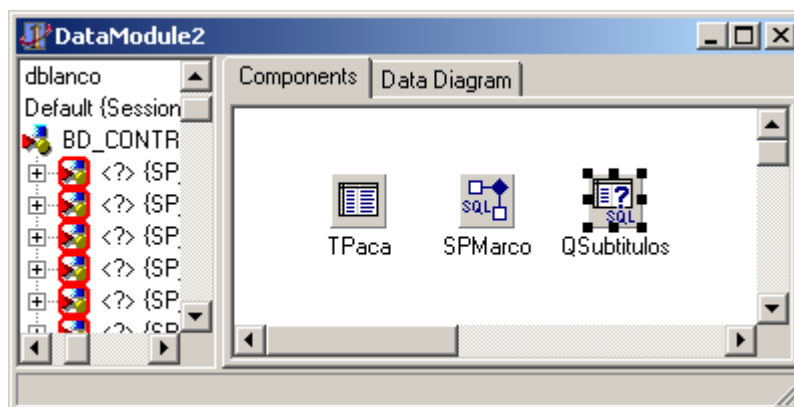


El Objeto de conexión que se configura dentro del código fuente de “SAPA” para acceder la base de datos es un *TDatabase*. Su configuración y definición puede hacerse gráficamente. En la siguiente ilustración se muestra el objeto gráfico *TDatabase* y sus propiedades de configuración:



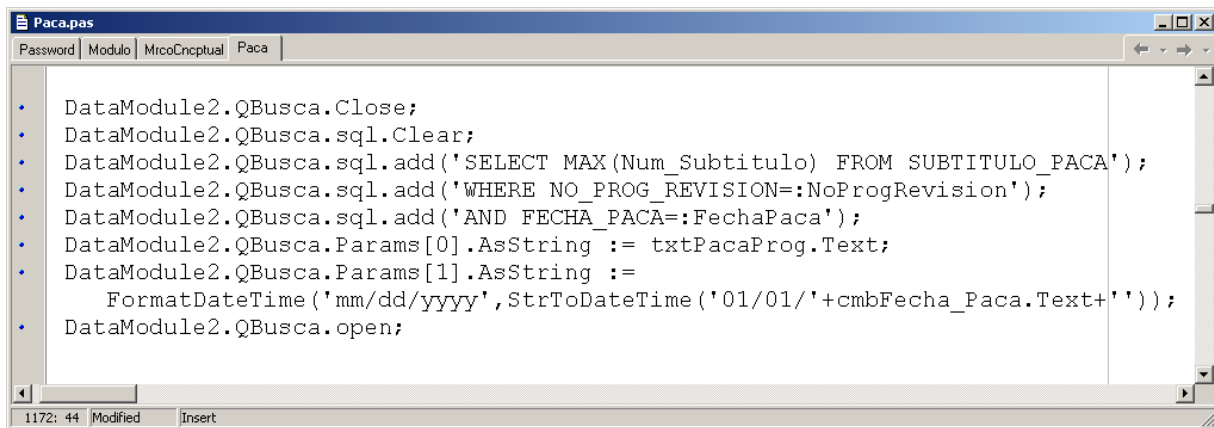
Acceso a Base de Datos

El acceso a los datos de la base se lleva a cabo en *Delphi 5.0* por medio de los siguientes objetos de datos:



- El objeto denominado *TPaca* es un ejemplo de un objeto de tipo *TTable* y sirve para acceder a los recursos de una sola tabla de manera rápida.
- El segundo objeto del gráfico, cuyo nombre es *SPMarco*, es un objeto de tipo *TStoredProc* y sirve para acceder a los datos que genera un procedimiento almacenado en la base de datos.
- El tercer objeto, *QSubtitulos* es un objeto de tipo *TQuery* y sirve para ejecutar código SQL capturado dentro de “SAPA”.

A continuación se muestra un ejemplo del código necesario para la ejecución de un query por medio del objeto *QBusca* de datos:



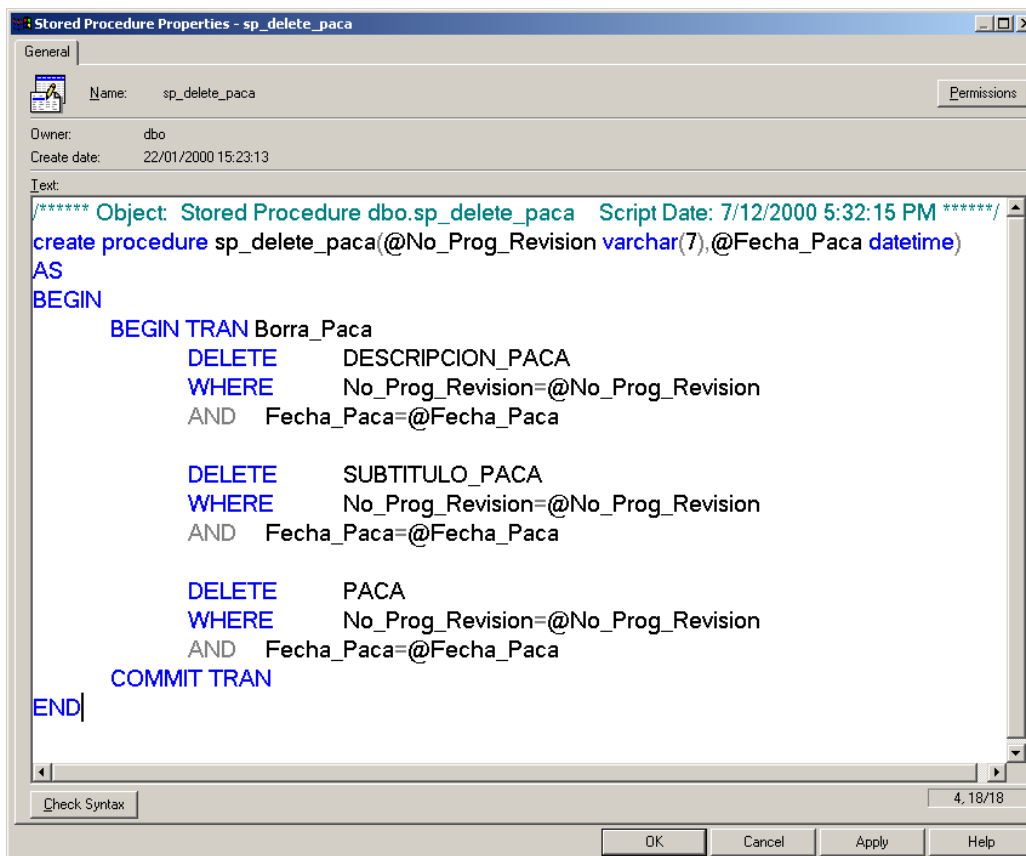
```
• DataModule2.QBusca.Close;
• DataModule2.QBusca.sql.Clear;
• DataModule2.QBusca.sql.add('SELECT MAX(Num_Subtitulo) FROM SUBTITULO_PACA');
• DataModule2.QBusca.sql.add('WHERE NO_PROG_REVISION=:NoProgRevision');
• DataModule2.QBusca.sql.add('AND FECHA_PACA=:FechaPaca');
• DataModule2.QBusca.Params[0].AsString := txtPacaProg.Text;
• DataModule2.QBusca.Params[1].AsString :=
  FormatDateTime('mm/dd/yyyy', StrToDateTime('01/01/'+cmbFecha_Paca.Text+''));
• DataModule2.QBusca.open;
```

Otro ejemplo de acceso a datos de la base se logra por medio de un *TStoredProc* al invocar un procedimiento almacenado en la base de datos:



```
• With DataModule2.SP1 do
  begin
  • try
  • Close;
  • StoredProcName:='sp_delete_paca';
  • Params.Clear;
  • Params.CreateParam(ftstring, '@Fecha_Paca', ptinput).asString:=
    FormatDateTime('mm/dd/yyyy', StrToDateTime('01/01/'+cmbFecha_Paca.Text+''));
  • Params.CreateParam(ftstring, '@No_Prog_Revision', ptinput).asString:=
    txtPacaProg.text;
  • ExecProc;
  • datamodule2.BD_CONTRALORIA.Commit;
  • ConsultaDatosPaca;
  • except
  • on Exception do
  • begin
  • datamodule2.BD_CONTRALORIA.Rollback;
  • end;
  • end;
  end;
```

El código fuente del procedimiento almacenado cuyo nombre es “*sp_delete_paca*” según se ve en el ejemplo es el siguiente:



```
/****** Object: Stored Procedure dbo.sp_delete_paca  Script Date: 7/12/2000 5:32:15 PM *****/
create procedure sp_delete_paca(@No_Prog_Revision varchar(7),@Fecha_Paca datetime)
AS
BEGIN
    BEGIN TRAN Borra_Paca
        DELETE     DESCRIPCION_PACA
        WHERE      No_Prog_Revision=@No_Prog_Revision
        AND        Fecha_Paca=@Fecha_Paca

        DELETE     SUBTITULO_PACA
        WHERE      No_Prog_Revision=@No_Prog_Revision
        AND        Fecha_Paca=@Fecha_Paca

        DELETE     PACA
        WHERE      No_Prog_Revision=@No_Prog_Revision
        AND        Fecha_Paca=@Fecha_Paca
    COMMIT TRAN
END
```

En los dos tipos de objetos anteriores, si se realiza una consulta de datos, los registros quedan dentro del objeto *TQuery* o *TStoredProc* y quedan disponibles para consulta de otros objetos como campos o listas o combos. Si la operación realizada es actualización, inserción o borrado también se ejecuta pero guarda ningún registro.



Bibliografía

BIBLIOGRAFÍA

DATE, C. J, *Introducción a los Sistemas de Bases de Datos*, Addison-Wesley, Iberoamericana, México, 1990.

GANE, Chris Sarson, Trish, *Structures System Analysis: Tools & Techniques*, Prentice Hall, E.U., 1979.

KENDALL, Kenneth E, *Análisis y Diseño de Sistemas*, Prentice Hall, México, 1991.

PRESSMAN, Roger, *Ingeniería del Software. Un Enfoque Práctico*, Mac Graw Hill, E.U.

MARTIN, James, *Organización de las Bases de Datos*, Prentice Hall, E.U, 1989.

Object-Oriented Application Analysis and Design for Java Technology (UML), Sun Microsystems, Inc., E.U., Revision B, March, 2000.

LARMAN, Craig, *UML y Patrones. Introducción al Diseño y Análisis Orientado a Objetos*, Prentice Hall, México, 1999.

OSIER, Dan, *Aprendiendo Delphi 2*, Sams Publishing y Prentice Hall, México 1996.