



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE INGENIERÍA

"IMPLEMENTACIÓN Y DESARROLLO DE
UN SISTEMA COMPUTACIONAL PARA LA
ADMINISTRACIÓN Y CONTROL DE CUENTAS
POR COBRAR DE LOS DESPACHOS JURÍDICOS"

T E S I S

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN COMPUTACIÓN

PRESENTA:

DAVID OCTAVIO DÍAZ TORRES

**DIRECTOR DE TESIS:
ING. JUAN MANUEL MARTÍNEZ VILLALOBOS**

2004



INTRODUCCIÓN	III
 CAPÍTULO 1. MARCO TEÓRICO	
1.1 Características de los despachos jurídicos.....	1
1.2 Administración de asuntos legales en los despachos jurídicos.....	3
1.3 Administración de asuntos contables en despachos jurídicos.....	5
1.4 Manejo y almacenamiento de información en los despachos jurídicos.....	7
 CAPÍTULO 2. IDENTIFICACIÓN DEL PROBLEMA	
2.1 Procedimientos actuales.....	9
2.2 Deficiencias actuales.....	13
2.3 Perspectivas.....	14
 CAPÍTULO 3. ANTECEDENTES DE LA METODOLOGÍA	
3.1 Métodos de Programación.....	15
3.2 El modelo de ciclo de vida clásico.....	16
3.3 Herramientas para el Análisis y Diseño de Software.....	25
 CAPÍTULO 4. ANÁLISIS	
4.1 Funcionamiento actual de la facturación de los Despachos Jurídicos.....	26
4.2 Requerimientos del usuario para optimizar el proceso de facturación.....	28
4.3 Herramientas de desarrollo existentes en el mercado que permitan optimizar el proceso de facturación.....	29
4.4 Diagnóstico de los recursos de cómputo con los que cuenta el cliente y requerimientos para la implementación de un sistema de facturación en un Despacho Jurídico.....	31
4.5 Importancia de la agilización del proceso de facturación para evitar tiempos no productivos del personal jurídico del Despacho.....	32
4.6 Diagramas de flujo de datos del proceso de facturación de los despachos jurídicos.....	33
4.7 Relevancia de la disponibilidad de consulta de los datos generados a través de reportes de información consolidada.....	36
4.8 Diagramas de flujo de datos del Sistema Jurídico de Cuentas por Cobrar (SJCC).....	38

CAPÍTULO 5. DISEÑO

5.1 Conceptos para el diseño de software.....	42
5.2 Diseño del sistema.....	47
5.3 Diccionario de datos.....	49

CAPÍTULO 6. DESARROLLO

6.1 Selección del Software.....	59
6.2 Etapa de desarrollo.....	65
6.2 Desarrollo del sistema.....	68

CAPÍTULO 7. PRUEBAS Y MANTENIMIENTO DEL SISTEMA

7.1 Pruebas.....	75
7.2 Seguridad.....	82
7.3 Mantenimiento del sistema.....	84

CONCLUSIONES.....	87
--------------------------	-----------

ANEXO 1. MANUAL DEL USUARIO.....	90
---	-----------

BIBLIOGRAFÍA.....	105
--------------------------	------------

INTRODUCCIÓN

En la actualidad el avance de las Tecnologías de Información tiene un ritmo bastante acelerado, por tal motivo el uso de este tipo de herramientas para el desarrollo de cualquier empresa es importante, cada día las empresas buscan nuevas formas de impulso tecnológico para el progreso de la organización.

No es muy común el uso avanzado de herramientas de cómputo en los despachos jurídicos, principalmente se debe a la falta de una capacitación adecuada para obtener un manejo óptimo y eficiente de los recursos informáticos y de esta manera obtener un desempeño integral de estos.

Los despachos jurídicos, tienen como propósito principal la atención de asuntos legales de todos sus clientes, entonces, es necesario tener un control apropiado de todos los datos que se generan por el manejo de los mismos, con el objetivo de optimizar los recursos humanos y financieros para el manejo apropiado de esta información.

La tesis consta de siete capítulos, los primeros tres capítulos sirven como introducción a la metodología empleada para el desarrollo del sistema y los restantes capítulos se enfocan al desarrollo de la metodología, siendo el modelo de ciclo de vida en cascada el utilizado, a continuación se explicará brevemente el contenido de cada capítulo.

En el capítulo uno se aborda de manera general el concepto de derecho y los diferentes tipos de despachos jurídicos que existen, también se describe brevemente el manejo y control de información de los asuntos legales y contables de los mismos.

El capítulo dos hace una descripción de la problemática actual de los despachos, explicando procedimientos y deficiencias de estos en cuanto al uso de herramientas tecnológicas para la administración y control de sus cuentas por cobrar, así como qué perspectivas de desarrollo esperan.

El capítulo tres se enfoca a describir qué es el ciclo de vida en cascada y las herramientas ocupadas para el análisis y diseño para el desarrollo del sistema de cuentas por cobrar de los despachos jurídicos.

En el capítulo cuatro se presenta el análisis del sistema de cuentas por cobrar de los despachos jurídicos, con la finalidad de identificar las necesidades que presentan los mismos, así como los requerimientos necesarios para la implementación de un buen sistema.

En el capítulo cinco se establece el diseño del sistema de cuentas por cobrar de los despachos jurídicos, en el cuál se reúnen las características obtenidas del análisis, para satisfacer las diferentes necesidades de los despachos en relación con su cobranza, también aquí se presentan el modelo entidad relación y el diccionario de datos.

En el capítulo seis se menciona el desarrollo del sistema de cuentas por cobrar de los despachos jurídicos, aquí se analizó diferentes tipos de herramientas de cómputo existentes en el mercado, eligiéndose la que mejor se acomode a las necesidades de desarrollo, señalando las características de funcionamiento y los requerimientos mínimos de instalación del sistema.

En el capítulo siete se describe y menciona las pruebas que se le efectuaron al sistema de cuentas por cobrar de los despachos jurídicos para su implementación, así como también el mantenimiento del mismo.

Este trabajo tiene como objetivo principal: a partir de las necesidades de administración y control de cobros de los despachos jurídicos, desarrollar un sistema de cómputo que controle y opere todas las actividades financieras referentes a los ingresos de un despacho jurídico; con una interfase amigable, de fácil uso, mostrando información concreta y precisa en cualquier momento de los estados financieros del despacho, así como de los clientes.

1. MARCO TEÓRICO

El Marco Teórico tiene como finalidad proporcionar un conjunto de elementos estructurados que permitan dar los primeros pasos en el proceso de toda investigación científica, a la vez sirve para delimitar las características propias del proyecto y funciona para ubicar al objeto de estudio en su exacta dimensión. Se presentan a continuación las referencias de estudio para el siguiente trabajo de investigación.

Se hace un estudio detallado de las características de los despachos jurídicos y la manera que estos organizan su contabilidad para poder determinar adecuadamente el desarrollo del sistema que se pretende establecer.

1.1 Características de los despachos jurídicos.

Definición de Derecho según la Real Academia Española: “Conjunto de principios, preceptos y reglas a que están sometidas las relaciones humanas en toda sociedad civil, y a cuya observancia pueden ser compelidos los individuos por la fuerza”. A continuación se enumeran y definen las características de los despachos de mayor relevancia jurídica en el mercado.

Definición de un despacho jurídico.

Se define el concepto de despacho jurídico como “el lugar o sitio, donde se ejerce el derecho en sus diversas formas o funciones de acuerdo a las necesidades que presente un cliente, que solicite o emplee servicios de un despacho”.

Despacho jurídico civil

Lugar donde se encargan de llevar asuntos de carácter civil, es decir, lo referente al derecho civil, el cual se establece como “el conjunto de preceptos que determina y regula las relaciones jurídicas entre los miembros de una familia y las que existen entre los individuos de una sociedad, para la protección de los intereses particulares, concernientes a sus personas y a sus bienes”

Despacho jurídico penalista

Lugar donde se encargan de llevar asuntos de carácter penal, es decir, lo referente al derecho penal, el cual se establece como “el conjunto de reglas establecidas por el Estado con el fin de unir al hecho del delito de la pena, como su consecuencia jurídica.

Despacho jurídico laboral

Lugar donde se encargan de llevar asuntos de carácter laboral, es decir, lo referente al derecho laboral, el cual se establece como “aquel que tiene por finalidad principal la regulación de las relaciones jurídicas entre empresarios y trabajadores, y de unos y otros con el Estado, en lo referente al trabajo subordinado y en cuanto atañe a las profesiones y a la forma de prestación de los servicios, también en lo relativo a las consecuencias jurídicas mediatas o inmediatas de la actividad laboral.”

Despacho jurídico fiscal

Se establece como una rama del derecho financiero y “es aquel que regula las relaciones entre el Erario público y los contribuyentes, a través de los impuestos de toda índole, las personas y bienes gravados, las exenciones especiales, formas y plazos del pago, a las multas u otras penas, o los simples recargos que corresponde aplicar por infringir preceptos relativos a declaraciones, trámites y vencimientos”.

1.2 Administración de asuntos legales en los despachos jurídicos.

Todo despacho debe tener una adecuada administración de los asuntos legales, ya que a partir de esto se optimizan las funciones del personal y se asignan apropiadamente las tareas a cumplir.

Manejo de Expedientes

La administración de los asuntos legales de los despachos jurídicos en el área de expedientes se lleva mediante la utilización de un archivero organizado en varias carpetas y separadores ordenados por el nombre del cliente que tiene asuntos legales en proceso. Cuando se desea consultar cuál es el avance de un asunto, se tiene que acudir a esta sección de archivo, seleccionar la gaveta que contiene los expedientes de cada cliente, y localizar el fólder que contiene el expediente. Se llega a dar el caso que un cliente desee consultar el estatus de todos sus asuntos, y es muy común que un abogado no esté enterado de todos los casos que se ventilan en el despacho, por lo que es necesario consultar todos los expedientes de ese cliente para cubrir el requerimiento de información.

Audiencias

En el manejo de las audiencias; se organizan las mismas anotando las fechas en una agenda general, después cada abogado transcribe la información a su propia agenda; posteriormente se tiene un día de la semana para asignar a cada uno de ellos, las audiencias que estarán a su cargo durante la siguiente semana.

Posteriormente, los abogados dictan un informe de las actividades que se cubrieron en cada audiencia, este forma parte de un recopilación diaria y general del despacho, que se plasma por escrito para su posterior consulta.

Organización de la información.

Actualmente en el área jurídica, los despachos utilizan las computadoras en su mayoría, como máquinas de escribir; limitándose a trabajar con un procesador de palabras y una impresora. En contadas ocasiones elaboran una hoja de cálculo para algunos formatos o

presentaciones de datos. En caso de que requieran imprimir en otro equipo, si es que no tienen una impresora disponible, graban la información en un disquete y trasladan la información a una computadora que tenga impresora.

En cuestión de intercambio de archivos, no cuentan con la debida protección para el ataque de virus informáticos.

1.3 Administración de asuntos contables en despachos jurídicos.

El manejo de información contable en cualquier empresa u organización es de vital importancia, en lo que respecta a los despachos jurídicos la administración adecuada de esta información determina la utilización de los recursos existentes de forma ordenada facilitando la creación de un plan de trabajo conveniente, ya que con esto se establecen las diferentes actividades que esto conlleva; el control de ingresos y egresos, los tipos de emisión de facturas (iguales mensuales, gastos incurridos u honorarios).

Control de Ingresos y Egresos.

Mensualmente se lleva a cabo un proceso mediante el cual se facturan todos los servicios a los clientes, los cuales incluyen, iguala mensual, gastos y honorarios; este proceso es con máquinas de escribir manuales o eléctricas, hojas de cálculo, calculadoras, libretas de registro o notas adicionales.

La herramienta más común en los despachos, para llevar la contabilidad y las relaciones de cuentas por cobrar, ingresos, cuentas por pagar, facturación etcétera; es por medio de una hoja de cálculo. Es frecuente el manejo de este tipo de herramienta para capturar los datos de facturación de todos los clientes, la labor se hace muy engorrosa y lleva demasiado trabajo, por consiguiente mucha pérdida de tiempo.

Para los egresos de las compañías es común que ellos emitan reportes de los gastos ocasionados en cierto periodo de tiempo, haciendo necesario tener un control apropiado de gastos efectuados por el despacho, tanto de los socios, abogados, pasantes o del despacho en general.

Formas de facturación (masiva, particular).

En lo que respecta a los ingresos, comúnmente estos son como se detalla a continuación.

1) **Iguales mensuales.** Consiste en que el despacho se encarga de todos los asuntos legales de una empresa por periodos de tiempos estipulados por ambas partes, realizando servicios como: consulta telefónica, revisión de documentos, atención de los trámites

legales ante tribunales, preparación de convenios, pláticas de conciliación, etcétera. Cada fin de mes se imprimen las facturas del siguiente período de facturación de cada cliente.

2) Gastos incurridos. Durante todo el desarrollo de un asunto legal, existen erogaciones en que incurren los abogados por trámites administrativos como pueden ser: copias fotostáticas, pagos de servicios, compensaciones, etcétera. Estos gastos son cubiertos inicialmente por el despacho, posteriormente se le cobra al cliente, teniéndose en algunos casos un incremento por un factor determinado. El abogado llena una forma de gastos en donde indica la fecha del gasto, el importe, que asunto trató, a que expediente corresponde y a que cliente se le cargará el importe. A fin de mes se realiza un reporte de los asuntos tratados y se le entrega al cliente.

3) Honorarios. Esta forma de ingreso consiste en una tarifa por hora que cada abogado tiene del tiempo de atención de algún asunto legal, el abogado estima qué cantidad de tiempo le llevó atender el asunto, la tarifa por hora y el resumen del asunto tratado. Para llevar un control de todos estos asuntos se imprime un reporte donde se especifica lo siguiente: concepto de honorarios, la persona que solicitó el servicio, abogado que atendió, la tarifa por hora, el tiempo de atención, el costo total del servicio; de todo esto se hace una suma total mensual de los servicios ofrecidos al cliente, llevándose un control de los gastos realizados en ese periodo de tiempo.

1.4 Manejo y almacenamiento de información en los despachos jurídicos.

Regularmente los despacho jurídicos tienen una administración deficiente de la información, por la cantidad de datos que se operan, es decir, lo llevan a cabo a través de archiveros, guardando toda la información de todos los asuntos que ellos manejan durante cierto periodo de tiempo, entonces se acumula una gran fuente de papelería que a la larga tiene que almacenarse en archivos muertos.

Cuando necesitan obtener cierta información de un cliente se procede a obtenerlo del archivero y seleccionarlo por el nombre del cliente o por el asunto que se trató, el manejo de expedientes es un poco engorroso de esta forma, ya que se acumula gran cantidad de papelería de todos los expedientes del despacho.

De igual manera se lleva un control general de la contabilidad de los despachos por medio de expedientes que se van acumulando a lo largo del tiempo, ya sea porque se factura a un cliente por asunto o por una pago, por ejemplo por iguala, que es cuando un cliente contrata al despacho para que lleve los asuntos de su empresa durante un trascurso de tiempo que se estipula entre el despacho y el cliente.

Es importante para los despachos el manejo adecuado de todos los datos y llevar un control adecuado de la administración de toda la información, ya sea por los asuntos legales o los asuntos contables del despacho.

En ocasiones resulta bastante complicado el manejo de tanta información, debido a que el desarrollo de un asunto legal puede tomar bastante tiempo y los despachos tienen que llevar un seguimiento detallado del mismo, entonces cuando se tengan que cobrar estos servicios se incluyan todos los gastos y cuentas que se hallan generado por dicho caso, o en su defecto ir cobrando cada vez que se lleve a cabo algún movimiento por dicho asunto (esto por honorarios o una iguala mensual).

Audiencias

En lo que respecta al manejo de la información de las audiencias, por lo regular ellos no tiene un control detallado de cómo y cuándo será dicho evento, sino que tiene una pizarra

de anotación donde se va acumulando la información que los mismos abogados van generando o una persona administrativa se encarga de llevar a cabo esta tarea.

Aquí también es importante conocer cuáles son las audiencias, de quién es el caso y dónde se llevará a cabo, esto para saber el tipo de cobro que se hará por gastos profesionales o por gastos de asesoría legal, que se deben considerar a la hora de hacer la facturación de los clientes.

Organización de la información

Para el manejo de la información no se cuenta con un adecuado control de expedientes, tanto contables o fiscales, entonces es importante llevar a cabo una metodología que permita tener una organización acorde con las necesidades del despacho.

Procedimientos (herramientas) convencionales.

Por lo general es común, que en muchos despachos sigan manejando máquinas de escribir como herramientas para la elaboración de facturas de la compañía, resulta muy difícil emitir facturas, ya que el tiempo hombre utilizado para esto es demasiado o es un poco complicado por el vaciado de información que se lleva a cabo, esto representa recursos desaprovechados, que al despacho le cuesta mucho tiempo y esfuerzo del personal administrativo, por consiguiente limita el desarrollo adecuado del despacho.

Cuando llegan a usar una herramienta computacional esto se realiza a través de una hoja de cálculo, que agiliza un poco la realización de la facturación, sin embargo no representa un cambio significativo en la disminución de tiempo hombre utilizado para la facturación de la compañía.

Por lo que respecta a la impresión de la factura en una hoja de cálculo, se tiene que adecuar el documento cada vez que se tenga que imprimir, esto debido a que no se cuenta con una forma estándar de impresión, con lo que limita aun más el manejo de esta herramienta de cómputo para emitir la facturación del despacho y por consiguiente desperdicio de recursos del mismo.

2. IDENTIFICACIÓN DEL PROBLEMA.

El desarrollo constante de las tecnologías de información es un punto importante a tratar por parte de cualquier empresa si desea sobresalir con respecto a la competencia, entonces es importante para los despachos jurídicos tomar ventaja de esto y aprovechar el crecimiento de las herramientas de tecnología de información.

La forma de administración de los cobros y cuentas por parte de los despachos jurídicos ha sido por mucho tiempo obsoleto, entonces al implementar un sistema de cómputo para agilizar sus trámites contables, se pretende optimizar el funcionamiento de los mismos.

La necesidad de estas compañías de consolidarse dentro del ámbito laboral es vital, de tal forma que la administración interna de la contabilidad es un punto primordial para el desarrollo sustentable de los despachos, esto en gran medida porque la mayoría de estas organizaciones no toma mucho en cuenta la optimización de los procesos administrativos, debido a que solamente organizan su contabilidad a través de una persona con conocimientos contables, pero que realmente no tiene la capacitación necesaria para lograr un adecuado manejo de los recursos tecnológicos.

Es importante para los despachos jurídicos establecer un adecuado manejo de su información, con el objetivo de hacer más eficaz su funcionamiento y lograr mayores beneficios para la compañía, para lograr esto es fundamental apoyarse en el desarrollo de las tecnologías de información, que permitan optimizar los recursos de la empresa, tanto en tiempo, como materiales y humanos.

2.1 Procedimientos Actuales

En la operación de un sistema manual de los despachos jurídicos se realiza la impresión de facturas en un tiempo sumamente mayor que mediante recursos de cómputo. La obtención de los conceptos de cobro y pago por parte del personal profesional del despacho conlleva cierto tiempo de ocupación de asuntos de control, tiempo que se podría considerar perdido pues se podría dedicar a los asuntos legales. El registro actual lleva

bastantes datos por cliente, por abogado, por fecha, por asunto (expedientes) y por evento o concepto.

La herramienta de software más utilizada para almacenar y realizar operaciones contables de los despachos jurídicos es una hoja de cálculo. Estos programas muestran un buen desempeño para realizar operaciones matemáticas; desgraciadamente tiene muchas limitantes no conocidas por los usuarios en comparación con las bases de datos. A continuación se listan varias situaciones relacionadas con el manejo de información contable por parte de los despachos jurídicos.

- Se utiliza una hoja de cálculo o un procesador de palabras para imprimir las facturas o recibos del despacho, esto se logra dando los espacios y márgenes necesarios para cuadrar la información con los campos que están preimpresos en la factura o recibo. Para facturaciones masivas se hace uso del comando “reemplazar” para sustituir la fecha en todo el documento, que varía en hojas según el número de facturas o recibos a imprimir. Si se van a imprimir 200 facturas, se editan 200 hojas.
- A discreción del usuario, éste acepta que se le realicen cobros atrasados o parciales de facturas o recibos de servicios ya devengados, además de que pudieran ser o no cobros relacionados con el orden cronológico en que fueron impresas las facturas.
- Se usa una hoja de cálculo para registrar las operaciones de facturación y pagos de los clientes, esto es, si el despacho tiene 70 clientes, se tienen 70 hojas de cálculo, lo se representa en la siguiente figura.

Díaz y Asociados S.C.

RFC: DIA-990617AZ
 Tulipanes 2100, Col. Napoles, Del. Benito Juárez
 México, Distrito Federal C.P.05400
 Tel. 57689000 Fax 57689000

Nº de factura: 777

FACTURA

Datos Fiscales del Cliente

Nombre	Oscar Acevedo Ugarte		
Dirección	Kansas No. 194, local A, Col. Napoles, Del. Benito Juárez		
Ciudad	de México	Estado D.F.	CP 45678
Teléfono	55366447	RFC:	AEU0690606TE6

Fecha	21-May-04
Nº Factura	777
Representante	

Cantidad	Concepto	Precio unitario	TOTAL
50	Servicios Profesionales Desarrollo del caso Acevedo vs. Castro concepto de honorarios profesionales que se le paga al despacho jurídico Díaz y Asociados	\$500	\$25,000

Subtotal	\$	25,000.00
IVA	\$	3,750.00
TOTAL	\$	28,750.00

Detalles de pago

En efectivo
 Con cheque
 Tarjeta de crédito

Nombre _____
 Número _____
 Caduca _____

CANTIDAD CON LETRA

Veintiocho mil setecientos cincuenta pesos 00/100 MN

Fig. 2.1.1 Formato de Factura Utilizado en los despachos jurídicos

- Se usa un block de notas para registrar a mano los pagos realizados por los clientes y los datos relacionados al pago, como son: fecha, forma de pago, datos de la forma de pago, importe, cliente, se muestra esto en la Figura 2.2.

Díaz y Asociados S.C.

Tulipanes 2100, Col. Napoles, Del Benito Juárez
México, Distrito Federal C.P. 05400
Tel. 5789000 Fax 5789000

Nº de informe **TIU**

INFORME DE GASTOS

Empleado

Nombre	Fernando Martínez	No. de Empleado:	34
Nº Seg. Social	2324234856	Puesto:	Abogado
Departamento	Jurídico	Superior:	Luis Sanchez

Período de pago

Desde	01/11/03
Hasta	01/12/03

Fecha	Cuenta	Descripción	Transporte	Combustible	Comidas	Otros	TOTAL
01/11/03	234234	Pagos	\$ 150.00	\$ 50.00	\$ 500.00	\$ 300.00	\$ 1,000.00
			\$ 150.00	\$ 50.00	\$ 500.00	\$ 300.00	
							Subtotal \$ 1,000.00
							Restar adelantos
							TOTAL \$ 1,000.00

Aprobado por

Luis Sanchez

Notas

Todos los cobros o pagos se realizan hasta que liquide su cuenta el cliente

Reembolso

Se requiere pago

Restar adelantos

TOTAL \$ 1,000.00

Fig. 2.1.2 Reporte de gastos de los despacho

2.2 Deficiencias Actuales

- No se mantiene un registro de las facturas generadas.
- Los datos que se desprenden de una factura solo pueden ser consultados leyendo la misma nuevamente, ya que no se almacenan los conceptos, los importes parciales, observaciones, quien las genero, fechas, etcétera.
- No existen datos informáticos para realizar consolidaciones de información a partir de las facturas impresas. Se deben de capturar nuevamente en otra hoja de cálculo para consultar: el importe de la facturación impresa, adeudos, pagos de clientes, etcétera.
- Si se requieren estados de cuenta de varios aspectos contables o financieros, simplemente se deben realizar varios procesos para obtenerlos.
- En todo momento este método de facturación esta ligado a la posibilidad de un error derivado del factor humano, lo cual se contrapone al uso de la computadora, que trata de minimizar la presencia de este aspecto.
- Definitivamente se subraya que el uso de una herramienta computacional se concibió primariamente para ayudar al humano a realizar operaciones complejas en un tiempo razonablemente mas breve, así mismo la búsqueda de la automatización de procesos tiene como objetivo limitar el “factor del error humano”; y la consecuencia de todo esto es el acceso a información veraz y oportuna que ayude a una correcta toma de decisiones.

2.3 Perspectivas

Es preponderante para un despacho mantener el control de los ingresos derivados de la operación del mismo, en función de tener un registro pleno y total de los adeudos y de los pagos realizados por el cliente. La seriedad y profesionalismo con que se lleva este asunto con los clientes debe ser clara, concisa y expedita para mantener una buena relación con los mismos. Se propone un sistema computacional que controle la emisión de facturas, el registro de pagos de clientes y la consolidación de conceptos de cobro por periodos definidos por el usuario, el Sistema Jurídico de Cuentas por Cobrar, que por facilidad del manejo de datos a partir de ahora le denominaremos SJCC, pretende facilitar el manejo de información que se genera a partir de la facturación del despacho, por ejemplo: estados de cuenta, detalle de conceptos agrupado en un reporte, ahorro de tiempo y optimización de recursos humanos y materiales, ya que el sistema pretende cubrir muchas áreas operativas de una administración contable.

El SJCC, buscará automatizar las operaciones de facturación y consolidación de información de un despacho jurídico. Mediante el uso de este sistema se pretende capturar los datos de los clientes en una base de datos, así mismo los importes y conceptos de facturación. Esta información se imprimirá masivamente o por evento de una manera más rápida, ya que no se requerirá volver a introducir los datos generales de la factura, sino se obtienen directamente de los datos previamente almacenados. Además, el proceso de generación de facturas, automáticamente generará registros que se almacenan en otras tablas, que serán utilizadas en otros aspectos relacionados; por ejemplo: para el registro de los pagos de las facturas, datos contables de información histórica de cada cliente, etcétera.

A su vez, un aspecto no contemplado por la dirección de los despachos jurídicos es la cuestión de seguridad, respaldo y confidencialidad de la información; todo esto se pretende este cubierto por el SJCC. En los siguientes capítulos se detallará específicamente cada módulo, de que consta y su funcionalidad.

3. ANTECEDENTES DE LA METODOLOGÍA

El desarrollo de una metodología adecuada sirve como fundamento para la implementación correcta de cualquier sistema de software, aquí se menciona la metodología ocupada para el SJCC.

3.1 Métodos de Programación.

La ingeniería del software deriva de la ingeniería de sistemas y de hardware, de esta manera podemos estipular que abarca un conjunto de tres elementos claves: métodos, herramientas y procedimientos, esto es lo que comúnmente se conoce como paradigmas de la ingeniería del software. La elección de un paradigma para la ingeniería del software se lleva a cabo de acuerdo con la naturaleza del proyecto y de la aplicación, los métodos y herramientas a usar, los controles y entregas requeridos.

Tres son los paradigmas que se han tratado ampliamente: el ciclo de vida clásico, construcción de prototipos y el modelo en espiral. El paradigma de ciclo de vida clásico exige un enfoque sistemático y secuencial del desarrollo del software que comienza en el nivel del sistema y progresa a través del análisis, diseño, codificación, prueba y mantenimiento.

El paradigma de construcción de prototipos es un proceso que facilita al programador la creación de un modelo del software a construir. El modelo puede tomar una de las tres formas siguientes: un prototipo en papel o un modelo basado en PC que describa la interacción hombre-máquina; un prototipo que implemente algunos subconjuntos de la función requerida del programa deseado, o uno existente que ejecute parte o toda la función deseada. Comparando ambos paradigmas determinamos qué el que mejor se ajusta a nuestras necesidades por sus características es el ciclo de vida clásico.

Resultados esperados.

El sistema pretende optimizar los recursos de administración en un despacho jurídico, de tal manera que no se desperdicien recursos humanos y materiales, con el manejo fácil de la información, que se genera a partir de la facturación del despacho.

3.2 El modelo de ciclo de vida clásico.

Podemos entender por el ciclo de vida clásico, como la sucesión de etapas por las que pasa el software desde que inicia un proyecto hasta la finalización del mismo.

Podemos encontrar diferentes tipos de modelos de ciclo de vida clásico, es decir, diversas formas de ver el proceso de desarrollo del software, cada uno lleva asociado una serie de métodos, herramientas y procedimientos que se deben usar a lo largo del proyecto. Para elegir un modelo de ciclo de vida clásico, debemos tomar en cuenta la naturaleza del proyecto, así como la aplicación, los métodos a usar, los controles y entregas requeridos.

Para nuestro caso nos centraremos en el ciclo de vida clásico en cascada o simplemente ciclo de vida clásico. Este es uno de los paradigmas antiguos mas utilizados en la Ingeniería del Software (IS). Este modelo se desarrolló a partir del ciclo convencional de ingeniería, es un ciclo de vida amplio, que incluye no sólo las etapas de ingeniería sino toda la vida del producto: las pruebas, el uso (la vida útil del software) y el mantenimiento, hasta que llega el momento de sustituirlo. Representado en la siguiente figura:

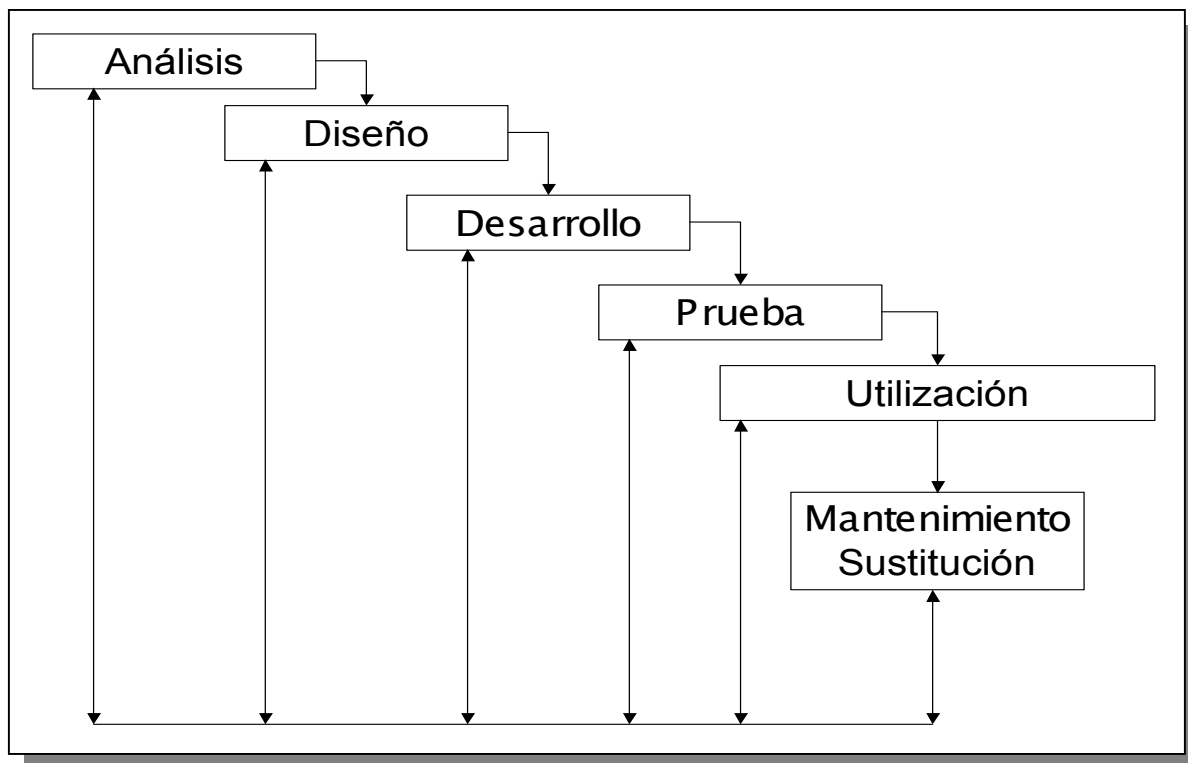


Fig. 3.1 Ciclo de vida en cascada

El ciclo de vida en cascada exige un enfoque sistemático y secuencial del desarrollo de software, que comienza en el nivel de la ingeniería de sistemas y avanza a través de fases secuenciales sucesivas. Estas fases son las siguientes:

Análisis. En esta parte se lleva a cabo la formulación de la problemática, definiéndose las características propias del sistema, significa la documentación general del sistema antes de implementar alguna acción correctiva del mismo, ocupando en este proceso herramientas de análisis como son los diagramas de flujo de datos.

Diseño. Aquí se definen los detalles fundamentales para el desarrollo del sistema, donde se encuentran incluidos los diagramas generales del sistema, el diccionario de datos y el diagrama entidad-relación.

Desarrollo. Esta etapa es el desarrollo en sí del sistema, realizando comparaciones con los lenguajes a ocupar para la implementación del software del sistema, con la finalidad de justificar porque se escogió dicha herramienta para la elaboración del sistema.

Pruebas. Es conjunto de actividades planeadas que se llevan a cabo sistemáticamente, con la finalidad de consolidar la información del propio sistema, en realidad es una verificación y validación de las actividades del software.

Mantenimiento. Tiene dos fases mantenimiento correctivo y mantenimiento preventivo, el primero es un diagnostico y corrección de errores del sistema y el segundo se enfoca a fallas determinadas por los usuarios del sistema.

Crecimiento o Sustitución. Es la ampliación del sistema o simplemente la sustitución del mismo por otro que presente mejores características de desarrollo.

Bases de datos relacionales

Una base de datos relacional es aquella en donde todos los datos visibles al usuario están organizados estrictamente como tablas de valores y en donde todas las operaciones de la base de datos operan sobre estas tablas.

Estas bases de datos son percibidas por los usuarios como una colección de relaciones normalizadas de diversos grados que varían con el tiempo.

El modelo relacional representa un sistema de bases de datos en un nivel de abstracción un tanto alejado de los detalles de la máquina. De hecho, el modelo relacional puede considerarse como un lenguaje de programación más bien abstracto, orientado de manera específica hacia las aplicaciones de bases de datos.

En términos tradicionales una relación se asemeja a un archivo, una entidad a un registro y un atributo a un campo. Pero estas correspondencias son aproximadas en el mejor de los casos.

Una relación no debe considerarse como "solo un archivo", sino más bien como un archivo disciplinado, siendo el resultado de esta disciplina una simplificación considerable de las estructuras de datos con las cuales debe interactuar el usuario, lo cual a su vez simplifica los operadores requeridos para manejar esas estructuras.

- Cada "archivo" contiene solo un tipo de registros.
- Los campos no tienen un orden específico, de izquierda a derecha.
- Los registros no tienen un orden específico, de arriba hacia abajo.
- Cada campo tiene un solo valor.
- Los registros poseen un campo identificador único (o combinación de campos) llamado llave primaria.

Así, todos los datos en una base de datos relacional se representan de una y solo una manera, a saber, por su valor explícito (esta se denomina en ocasiones "principio básico del modelo relacional"). En particular, las conexiones lógicas dentro de una relación y entre las relaciones se representan mediante esos valores; no existen "ligas" o apuntadores visibles para el usuario, ni ordenamientos visibles para el usuario, ni grupos repetitivos visibles para el usuario, etcétera.

El Modelo Entidad-Relación

El modelo Entidad-Relación (E/R) es un modelo conceptual de datos, que sirve como herramienta para el modelo relacional de bases de datos, principalmente pretende dotar de significado a las estructuras de datos. EL modelo E/R se ha convertido en un estándar para el diseño de las bases de datos relacionales. Actualmente es utilizado por la gran mayoría de las metodologías de diseño de sistemas para el diseño de datos.

Entonces podemos entender al modelo entidad-relación de acuerdo a sus características:

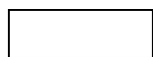
- Solo intuye la existencia de los datos.
- No le interesa lo que se hace con ellos.
- No se enfoca a aplicaciones en particular, esto debido a que se incorporan todos los datos del sistema.
- No tiene ninguna relación con los sistemas operativos y las bases de datos concretas.
- No es relevante las restricciones de almacenamiento, espacio, tiempo o ejecución.
- Evoluciona abiertamente con el sistema.

En todo caso lo que el modelo entidad-relación quiere dar es una perspectiva del mundo real de la forma más natural posible, esto con el uso de relaciones y objetos, no tiene nada que ver con el software o el hardware usados para su implementación, lo que se busca al implementarlo es mantener las propiedades adecuadas de las bases de datos.

Elementos de un modelo entidad-relación

- *Entidad*: Una entidad es un objeto que existe y puede ser distinguido de otro objeto. Una entidad puede ser concreta (un libro, un automóvil etc.) o abstracta (fecha, edad, etc.).
- *Atributo*: Una entidad se distingue de otra porque posee ciertas características que la hacen única. A estas características se les conoce como atributo. El rango de valores validos para un atributo determinado será conocido como dominio del atributo.
- *Entidades Fuertes*: Son aquellas en las que existen los atributos necesarios para formar una llave primaria.
- *Entidades Débiles*: No poseen los atributos necesarios para conformar una llave primaria.
- *Súperllaves*: Es un conjunto de atributos mediante los cuales es posible reconocer a una entidad. Este tipo de llaves contiene comúnmente atributos ajenos; es decir; atributos que no son indispensables para llevar a cabo el reconocimiento del registro.
- *Llave candidato*: Son aquellas súperllaves que no contienen atributos ajenos; es decir, aquellos conjuntos de atributos que no tienen un subconjunto menor que pueda considerarse como súperllave.
- *Llave primaria*: Es aquella llave que el diseñador de la base de datos selecciona entre las llaves candidatos encontradas.
- *Llave Foránea*: Son aquellos atributos de una entidad que son llave primaria en otra entidad.
- *Relación*: Se define como una asociación de diversas entidades, la representación gráfica es mediante un rombo.

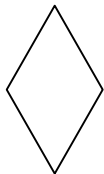
- *Conjunto de Relaciones:* Son todas aquellas relaciones que existen en un conjunto de entidades.
- *Dimensión de una relación:* Es el número de entidades que se pueden presentar en una relación, la dimensión mas general es la binaria, es decir en la que se relacionan dos entidades.
- *Clave de una relación:* Se denomina al producto cartesiano de las llaves primarias de las entidades asociadas, y también puede ser por los atributos que forman parte de esa relación.
- *Diagramas de entidad-relación:* Son esquemas que nos permitan representar conjunto de entidades y sus relaciones mediante la siguiente simbología:



Conjunto de entidades



Atributos




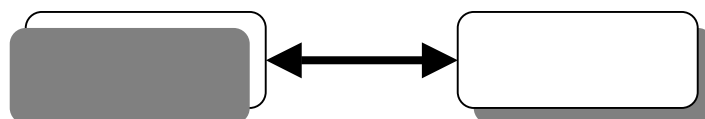
Conjunto de relaciones



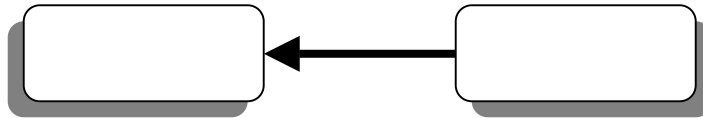
Conexión

- *Grado de relación:* Es la forma en como se relacionan las entidades y se presentan tres formas posibles:

- 1:1  (una a una): A cada atributo de una entidad le corresponde uno y sólo un atributo de otra entidad y viceversa.



- 1:m ← o m:1 → (una a muchas o muchas a una): Un atributo de una primera entidad puede corresponderle uno o varios atributos de una segunda entidad, y por consiguiente a la segunda entidad sólo le corresponde uno y sólo un atributo de la primera entidad.



- n:m — (muchas a muchas): A uno o más atributos de una primera entidad le pueden corresponder uno o más atributos de una segunda entidad y viceversa.



Normalización

En términos más sencillos la normalización trata de simplificar el diseño de una base de datos, a través de la búsqueda de la mejor estructuración que pueda utilizarse con las entidades involucradas en ella.

Pasos de la normalización:

- Descomponer todos los grupos de datos en registros bidimensionales.
- Eliminar todas las relaciones en la que los datos no dependan completamente de la llave primaria del registro.
- Eliminar todas las relaciones que contengan dependencias transitivas.

La teoría de normalización tiene como fundamento el concepto de formas normales; se dice que una relación está en una determinada forma normal si satisface un conjunto de restricciones.

Reglas de Normalización

Primera Forma Normal. Una relación R se encuentra en Primera Forma Normal (1FN) si y solo si por cada renglón columna contiene valores atómicos.

Abreviada como 1FN, se considera que una relación se encuentra en la primera forma normal cuando cumple lo siguiente:

- Las celdas de las tablas poseen valores simples y no se permiten grupos ni arreglos repetidos como valores, es decir, contienen un solo valor por cada celda.
- Todos los ingresos en cualquier columna(atributo) deben ser del mismo tipo.
- Cada columna debe tener un nombre único, el orden de las columnas en la tabla no es importante.
- Dos filas o renglones de una misma tabla no deben ser idénticos, aunque el orden de las filas no es importante.

Por lo general la mayoría de las relaciones cumplen con estas características, así que podemos decir que la mayoría de las relaciones se encuentran en la primera forma normal.

Como esta relación maneja valores atómicos, es decir un solo valor por cada uno de los campos que conforman a los atributos de las entidades, ya se encuentra en primera forma normal.

Segunda Forma Normal. Para definir formalmente la segunda forma normal requerimos saber que es una *dependencia funcional*: Consiste en edificar que atributos dependen de otro(s) atributo(s).

Una relación R está en Segunda Forma Normal (2FN) si y solo si está en 1FN y los atributos auxiliares dependen funcionalmente de la llave primaria.

Una relación se encuentra en segunda forma normal, cuando cumple con las reglas de la primera forma normal y todos sus atributos que no son llaves dependen por completo de la llave primaria. De acuerdo con esta definición, cada tabla que tiene un atributo único como llave primaria, esta en segunda forma normal.

Nótese que las llaves primarias estarán representadas con doble cuadro, las flechas nos indican que de estos atributos se puede referenciar a los otros atributos que dependen funcionalmente de la llave primaria.

Tercera Forma Normal. Una relación R está en Tercera Forma Normal (3FN) si y solo si esta en 2FN y todos sus atributos no primos dependen no transitivamente de la llave primaria.

Consiste en eliminar la dependencia transitiva que queda en una segunda forma normal, en pocas palabras una relación esta en tercera forma normal si está en segunda forma normal y no existen dependencias transitivas entre los atributos, nos referimos a dependencias transitivas cuando existe más de una forma de llegar a referenciar a un atributo de una relación.

Forma Normal de Boyce Codd. Uno o más atributos que, de manera funcional, determinan otro atributo o atributos. En la dependencia funcional $(A,B) \twoheadrightarrow C$, (A,B) son los determinantes.

Una relación R esta en Forma Normal de Boyce Codd (FNBC) si y solo si cada determinante es una llave candidato.

Denominada por sus siglas en ingles como BCNF, una tabla se considera en esta forma si y sólo si cada determinante o atributo es una llave candidato.

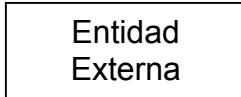
Obsérvese que a diferencia de la tercera forma normal, agrupamos todas las llaves candidato para formar una global las cuales hacen referencia a los atributos que no son llaves candidato.

3.3 Herramientas para el Análisis y Diseño de Software

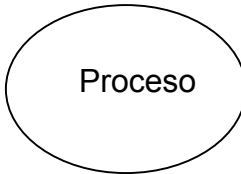
Diagrama de Flujo de Datos

El diagrama de flujo de datos (DFD) es una técnica que representa el flujo de la información y las transformaciones que se aplican a los datos al moverse desde la entrada hasta la salida.

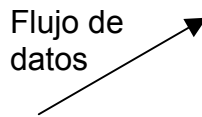
Simbología del Diagrama de Flujo de Datos



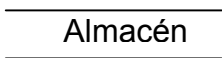
Se considera como alguien o algo que se produce o se consume fuera de los límites del sistema.



Aquello donde se transforma la información, es decir una función dentro de los límites del sistema



Se representa por medio de flechas y curvas El inicio de la flecha indica hacia donde se moverán los datos es decir, la dirección del flujo de datos.



Es donde se almacenan los datos para ser utilizados después en uno o más procesos y es representado entre dos líneas paralelas

4. ANÁLISIS

El desarrollo de esta etapa tiene como finalidad proporcionar las características propias del sistema, con el fin de obtener la documentación general, así implementar una acción correctiva sobre el mismo.

Para desarrollar esta etapa es necesario tomar en cuenta los siguientes puntos:

- Tratar de identificar las necesidades propias del cliente, esto a través de una adecuada comunicación entre el cliente y el desarrollador.
- La asignación apropiada de todos los recursos existentes para el desarrollo del sistema.

A continuación se procede a elaborar el análisis del sistema con los pasos anteriormente establecidos.

4.1 Funcionamiento actual de la facturación de los Despachos Jurídicos.

El procedimiento actual que se lleva a cabo, es registrando los eventos que van a generar obligaciones a los clientes de un despacho, los cuales se dividen en 4 segmentos: gastos erogados, honorarios devengados, iguala mensual y consultoría legal por tiempo.

Cada uno de estos segmentos se registra a mano en libretas o listas indicándose los siguientes datos:

- Gastos Erogados

Formato preimpreso de vales de gastos

Captura manual

Datos solicitados:

Fecha, nombre del abogado, importe en letra, importe en número, concepto del gasto, nombre del cliente, asunto o expediente de referencia, autorización.

- Honorarios devengados
Lista de registro actividades
Captura manual
Datos solicitados:
Fecha, nombre del abogado, importe en número, actividad, nombre del cliente.
- Iguala Mensual
Lista de registro de igualas por cliente
Captura manual
Datos solicitados:
Cliente, importe.
- Consultaría legal por tiempo
Lista de registro de consultoría
Captura manual
Datos solicitados:
Fecha, nombre del abogado, tiempo de consulta, actividad, tarifa, nombre del cliente, importe total.

En cualquier momento del mes se realiza una factura, siempre y cuando haya algo que necesite ser cobrado, por lo que se consultan los formatos previamente descritos y se transcribe la información con una máquina de escribir a una factura, sumando los importes y marcándolos como facturados. Con relación a las igualas, estas son facturadas mensualmente, por lo que cada mes se emiten todas las facturas de los clientes que se les cobra mediante una iguala mensual.

Después se registran en listas los importes de las facturas emitidas para obtener datos como: facturación mensual, totales facturados por cliente, se registran los pagos realizados en listas de control, en estas se anotan los siguientes datos: Fecha, número de factura, instrumento de pago, importe del pago, cliente, saldo, banco.

4.2 Requerimientos del usuario para optimizar el proceso de facturación.

El despacho requiere la automatización del proceso de registro de eventos que crean cargos a los clientes y que automáticamente se generen e impriman las facturas con los importes acumulados, y/o se realicen impresiones masivas de facturas en las que automáticamente se obtengan los conceptos de cobro y hojas de detalle de los eventos a cobrar.

A su vez es necesario tener un registro de los pagos realizados por los clientes, mantener al día los saldos de los mismos y los estatus de pago de cada factura emitida.

Todo esto se debe de llevar a reportes de información consolidada de ingresos mensuales, facturas emitidas, saldos de clientes, etcétera.

4.3 Herramientas de desarrollo existentes en el mercado que permitan optimizar el proceso de facturación.

Actualmente existen en el mercado programas de cómputo para realizar el proceso de facturación de una empresa, pero en un esquema más generalizado y enfocado a negocios que requieren registros transaccionales de mayor volumen. A su vez que no llevan un registro de las actividades o eventos que generan cargos o si los llevan es en un esquema de compraventa de bienes y en cuestión de servicios, no divide los eventos por segmentos de cobro, etcétera. Estos programas están enfocados más bien hacia un perfil de llevar en computadora los registros contables de la empresa.

Ejemplo de este tipo de programas tenemos:

- Sistema Administrativo Empresarial (SAE) de la compañía ASPEL, el cuál es un sistema de administración de ventas que permite controlar el ciclo de todas las operaciones de compra-venta de la organización.
- AdminPaq de Computación en Acción (Compac), que permite administrar el proceso comercial, pues controla a detalle las operaciones de compras, ventas, inventarios, clientes y proveedores. Además maneja múltiples almacenajes y monedas. Realiza reportes a la medida con su hoja electrónica.
- Macro Pro Integral y Macro Pro Comercial de Macro Pro, van al nicho de ventas a menudeo, como comercializadoras, distribuidores y empresas de servicio. Integra contabilidad, ventas, nómina, compras, inventarios y punto de venta. La única diferencia es que la segunda no incluye nómina.
- Administrador 2000(A2) de Sistemas Estratégicos, ayuda a la empresa a controlar y agilizar sus procesos y operaciones de compra-venta y se compone de cinco módulos: facturación, inventarios, cuentas por cobrar, compras y cuentas por pagar. Es un sistema multiempresa, multiusuario, multialmacén y multimoneda.

- ContaPlus de Grupo SP, entre sus bondades destacan el ser multiempresa, monopuesto y multiusuario; enlaza con AdminPlus Profesional y TPVPlus Profesional; genera nuevos tipos de ISR, tiene módulo de cheques y para análisis de impuestos; permite llevar contabilidad presupuestaria y por departamento, además de contener gráficos y enlace a Excel.

4.4 Diagnóstico de los recursos de cómputo con los que cuenta el cliente y requerimientos para la implantación de un sistema de facturación en un Despacho Jurídico.

En los despachos jurídicos actualmente usan diversas herramientas para llevar a cabo las tareas de facturación y cobranza. Comúnmente utilizan máquinas de escribir en la elaboración de las facturas y/o hojas de cálculo (Microsoft Excel, Lotus 123, Microsoft Works, etcétera) en donde ajustan los márgenes y espacios necesarios para centrar los campos (datos del cliente, importes, conceptos, fechas, etcétera) a imprimir en los formatos correspondientes. A su vez llenan hojas de cálculo con los consolidados de la facturación emitida. Estos métodos se ajustan al equipamiento actual, es decir, dependiendo de los recursos con que cuenten, llevan a cabo esta tarea.

Los puntos a recalcar al utilizar estos procedimientos son:

- Utilización de máquinas de escribir mecánicas o eléctricas.
- Siempre que se emite una factura, hay que meter los datos del cliente por cada factura realizada.
- No se lleva control de emisión de facturas, ni se consolida la información.

En el caso de las hojas de cálculo, se corre el riesgo de modificar el formato de impresión o los campos a escribir, ya que no se puede limitar fácilmente el acceso a los datos, se duplica la información al no tener normalizada en ninguna forma el almacenamiento de datos; además llega el momento en que se dificulta el manejo de hojas de cálculo con miles de celdas o cientos de columnas, con la correspondiente sobrecarga de los recursos de la computadora, volviéndola lenta en su operación.

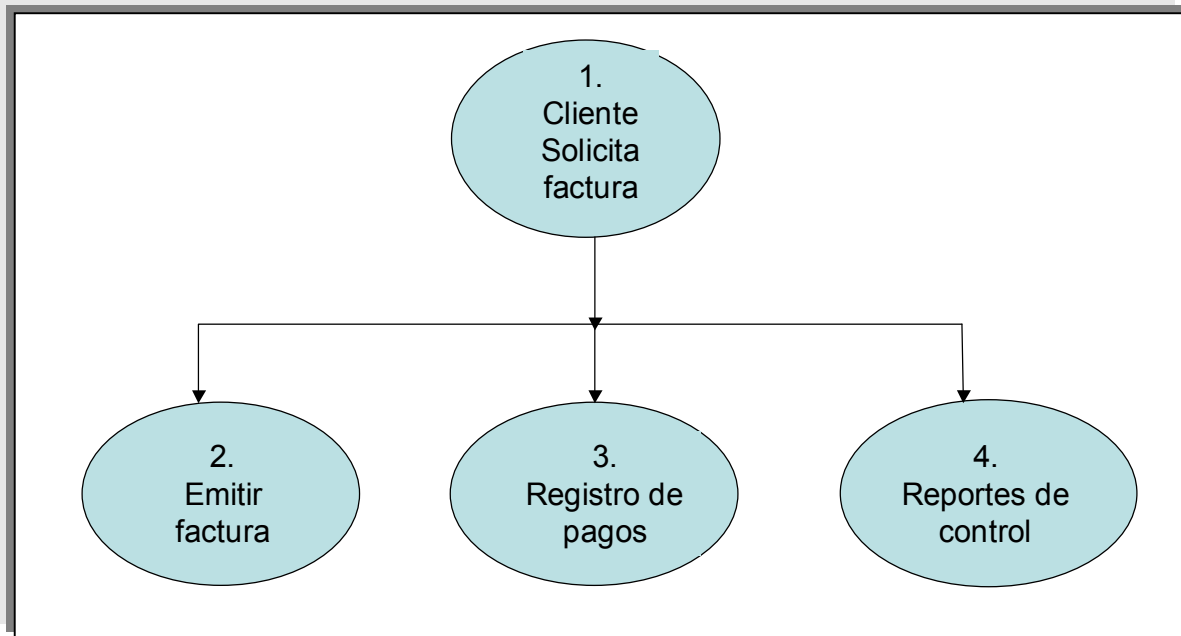
4.5 Importancia de la agilización del proceso de facturación para evitar tiempos no productivos del personal jurídico del despacho.

Es necesario que se evite asignar trabajo administrativo al personal jurídico del despacho, ya que estos tienen como principal función atender los asuntos legales que tienen encomendados. El llenado de los formatos para el registro de actividades tales como: vales de gastos (fotocopias, gastos de representación, viáticos, etcétera.) registros de tiempos (llamadas telefónicas, audiencias, elaboración de escritos, informes, etcétera.), y el consiguiente llenado de datos en estos registros de datos del cliente, fechas, asuntos relacionados, expedientes jurídicos, actores que intervienen, etcétera, generan tiempos no productivos que limitan el adecuado funcionamiento del despacho.

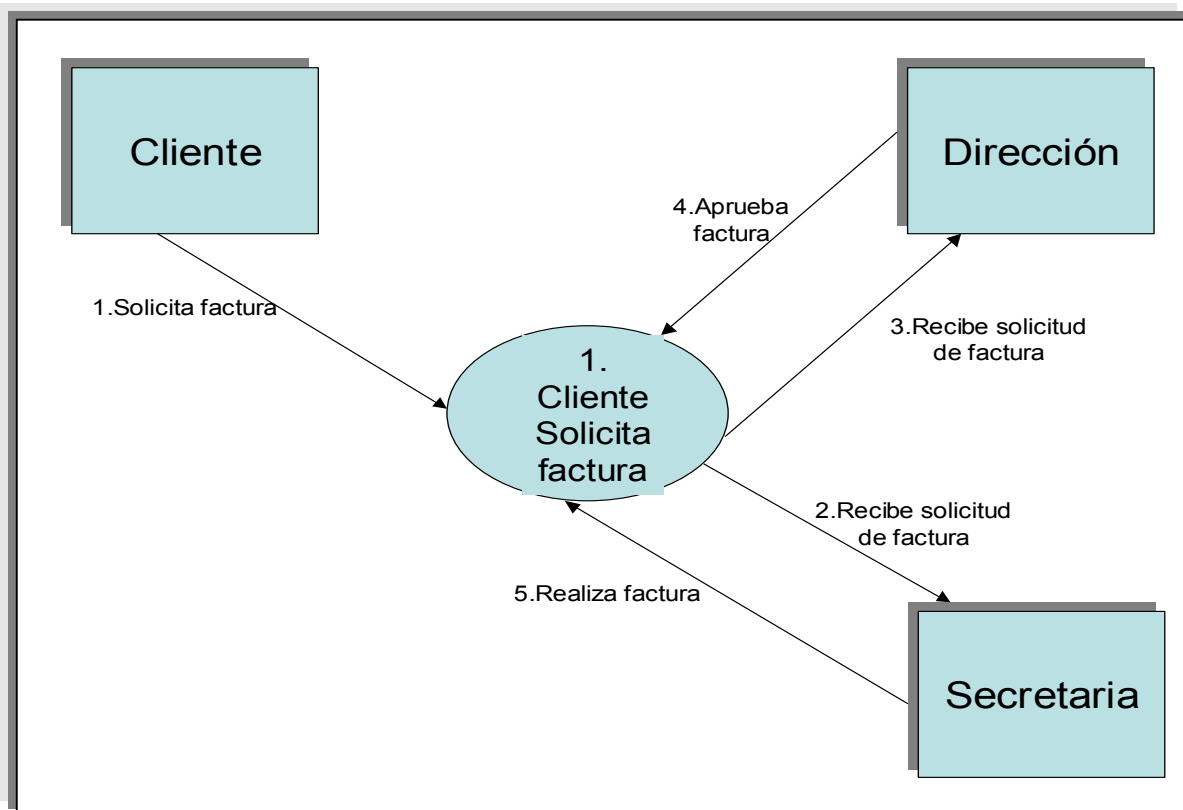
El uso de una herramienta computacional que agilice el proceso de registro de actividades administrativas y que tenga una operación intuitiva con una interfase fácil de manejar es el objetivo. Esto permite una mayor colaboración del personal y que el registro de actividades sea más rápido, minimizando problemas que se llegan a presentar: registro de datos erróneos, omisión de eventos, actividades sin pago en el periodo en que se incurren debido a que se presentan fuera del término estipulado para su presentación, duplicidad en el registro con el resultado de una mala imagen hacia el cliente.

4.6 Diagramas de flujo de datos del proceso de facturación de los despachos jurídicos.

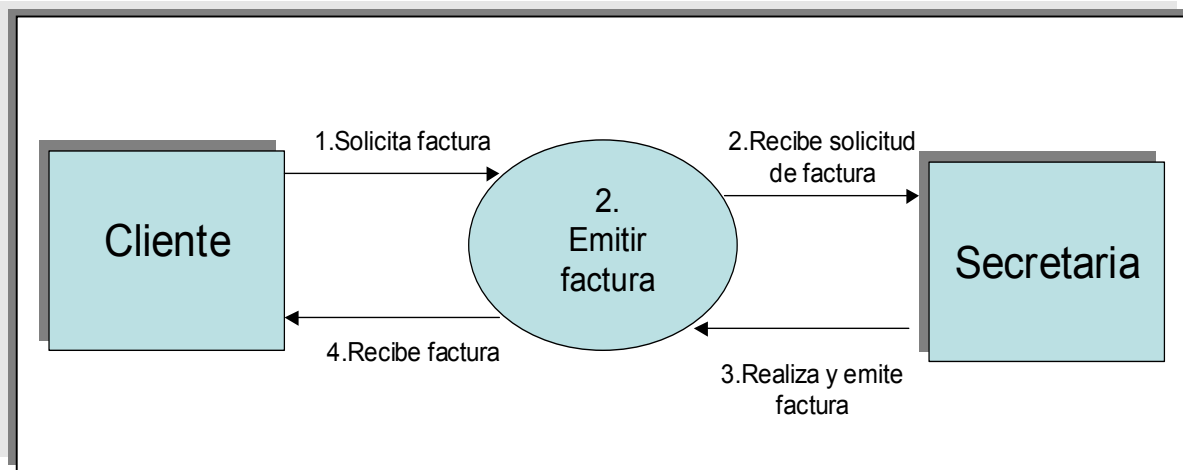
4.6.1 Diagrama de flujo de datos general del proceso de Facturación de los Despachos Jurídicos.



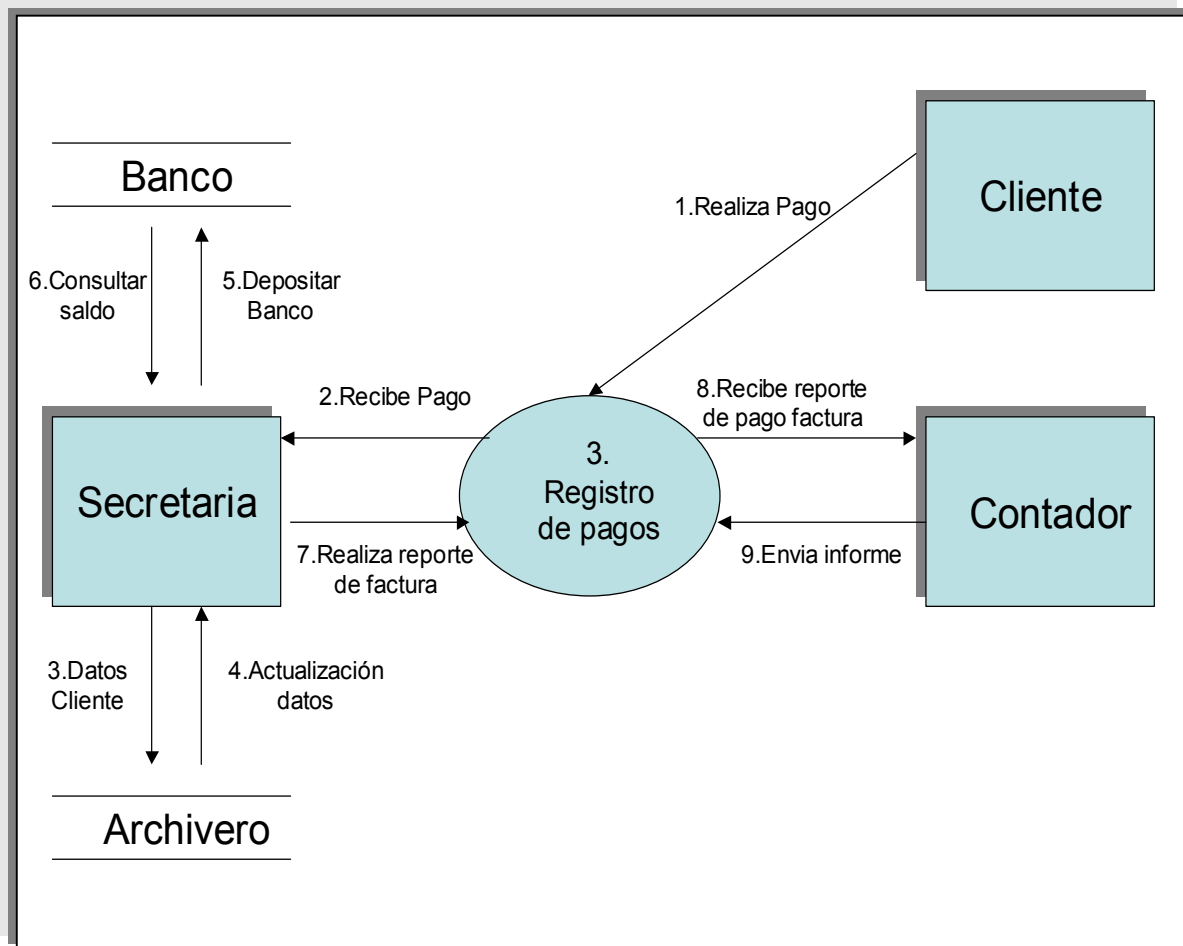
4.6.2 Diagrama de flujo de datos del proceso de solicitar una factura.



4.6.3 Diagrama de flujo de datos del proceso de emitir una factura.



4.6.4 Diagrama de flujo de datos del proceso de registro de pagos.



4.6.5 Diagrama de flujo de datos del proceso de reportes de control.



4.7 Relevancia de la disponibilidad de consulta de los datos generados a través de reportes de información consolidada.

Una de las principales funciones de las herramientas computacionales es la ayuda que generan para la toma de decisiones debido a que procesan datos e información a mayor velocidad que con su contra parte manual.

En los despachos jurídicos se requiere tener información consolidada en diversos aspectos, entre otros: importe de la facturación mensual emitida, cobros realizados en el mes, facturación acumulada por cliente, estados de cuenta del cliente, tarifas de las actividades facturadas, productividad por abogado, relación de cartera vencida y facturación en periodos de tiempos, etcétera.

El uso de herramientas de cómputo, es de gran ayuda para cualquier empresa ya sea un despacho, o algún otra organización, y conforme avanza la tecnología se notará más ampliamente el desarrollo de las herramientas computacionales y de esta manera podemos determinar la manera de procesar los datos que se van a ocupar en el momento adecuado y de una manera precisa, entonces el impulso y uso de este tipo de herramientas es muy útil para compañías como los despachos jurídicos.

La importancia de tener la información en el momento preciso, es un tópico muy buscado en el desarrollo de tecnología actual, empresas de tecnologías de información se empeñan en desarrollar mejores técnicas para que los datos de una empresa sean más concretos y más accesibles de manera que inviertan en este rubro para poder lograr este tipo de objetivos tecnológicos.

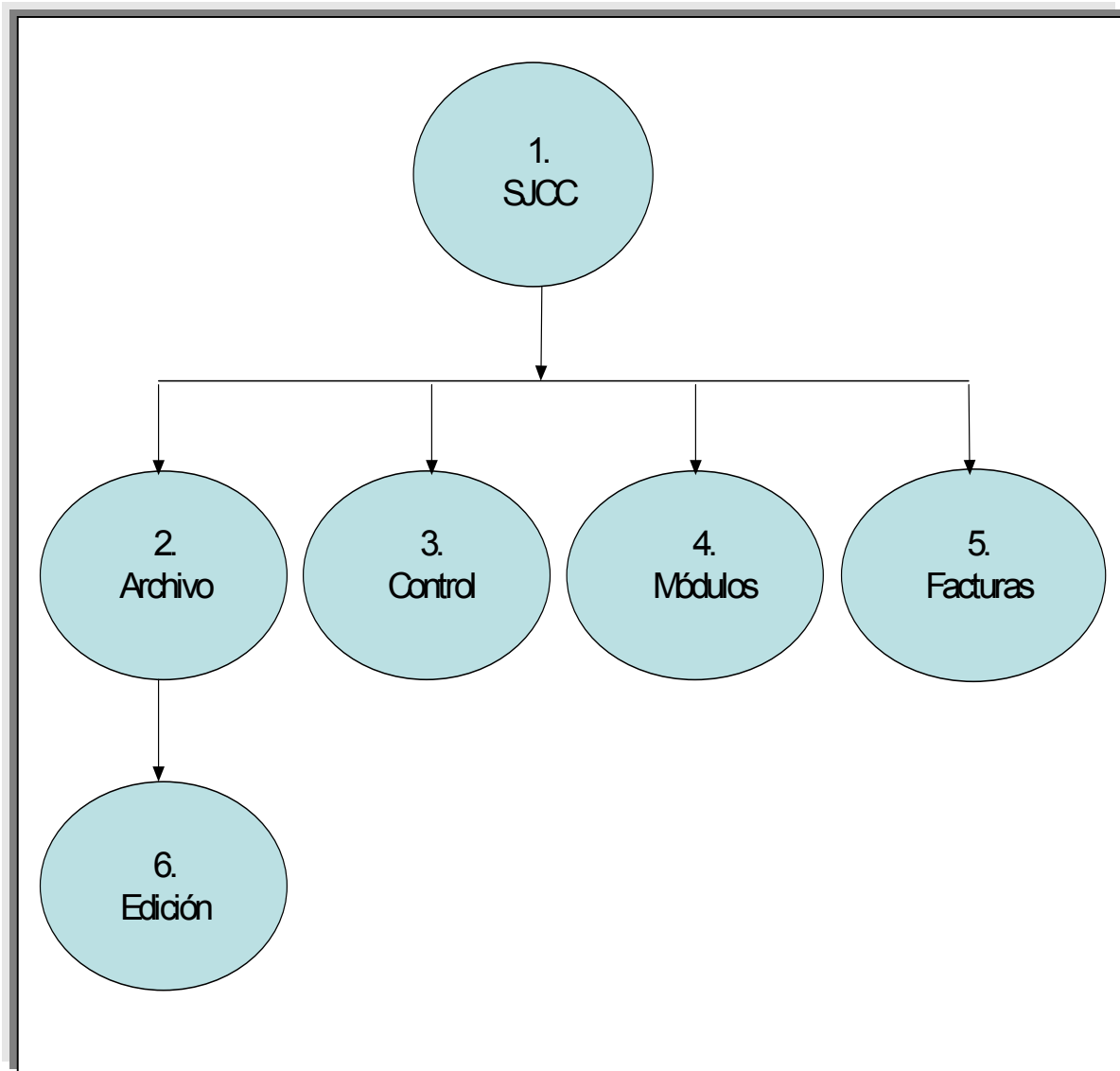
Podemos decir que cuando se desarrolla un sistema computacional, se hace con la finalidad de optimizar el manejo de la información de una empresa, por ejemplo, en el SJCC los despachos manejarán la información contable para ahorrar recursos tales como tiempo, dinero y de material, por esto la importancia de diseñar un sistema con este tipo de características, que facilite la administración de los despachos y la optimización de los recursos.

Considerando lo mencionado anteriormente se puede afirmar que el desarrollo de un sistema como el SJCC es de gran ayuda para la dirección de una empresa, porque optimiza la administración del mismo y permite distribuir los recursos de la compañía de manera que sean aprovechados en su totalidad.

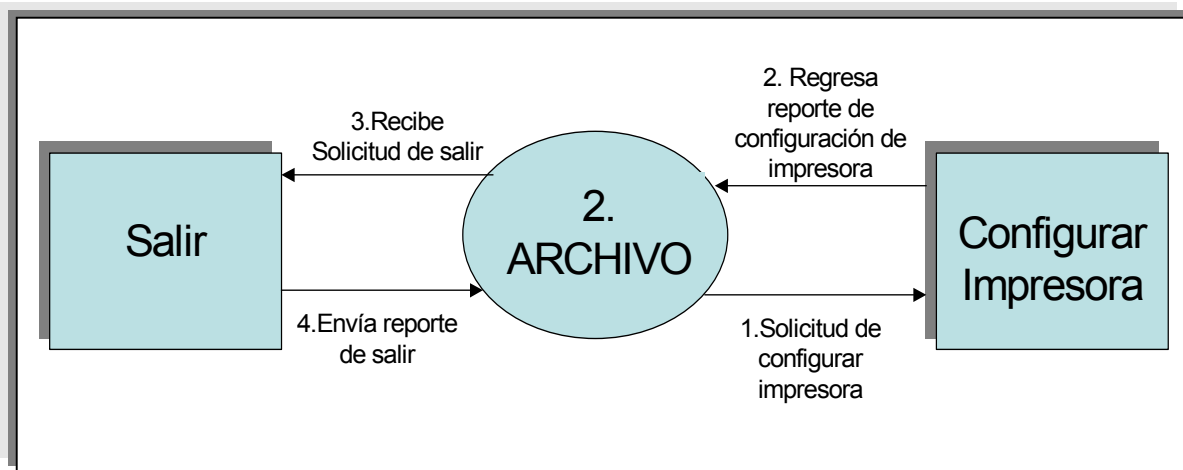
Entonces de acuerdo con los datos recopilados anteriormente se presentan los diagramas de flujo de datos que se crearon para el desarrollo del Sistema Jurídico de Cuentas por Cobrar (SJCC).

4.8 Diagramas de flujo de datos del Sistema Jurídico de Cuentas por Cobrar (SJCC).

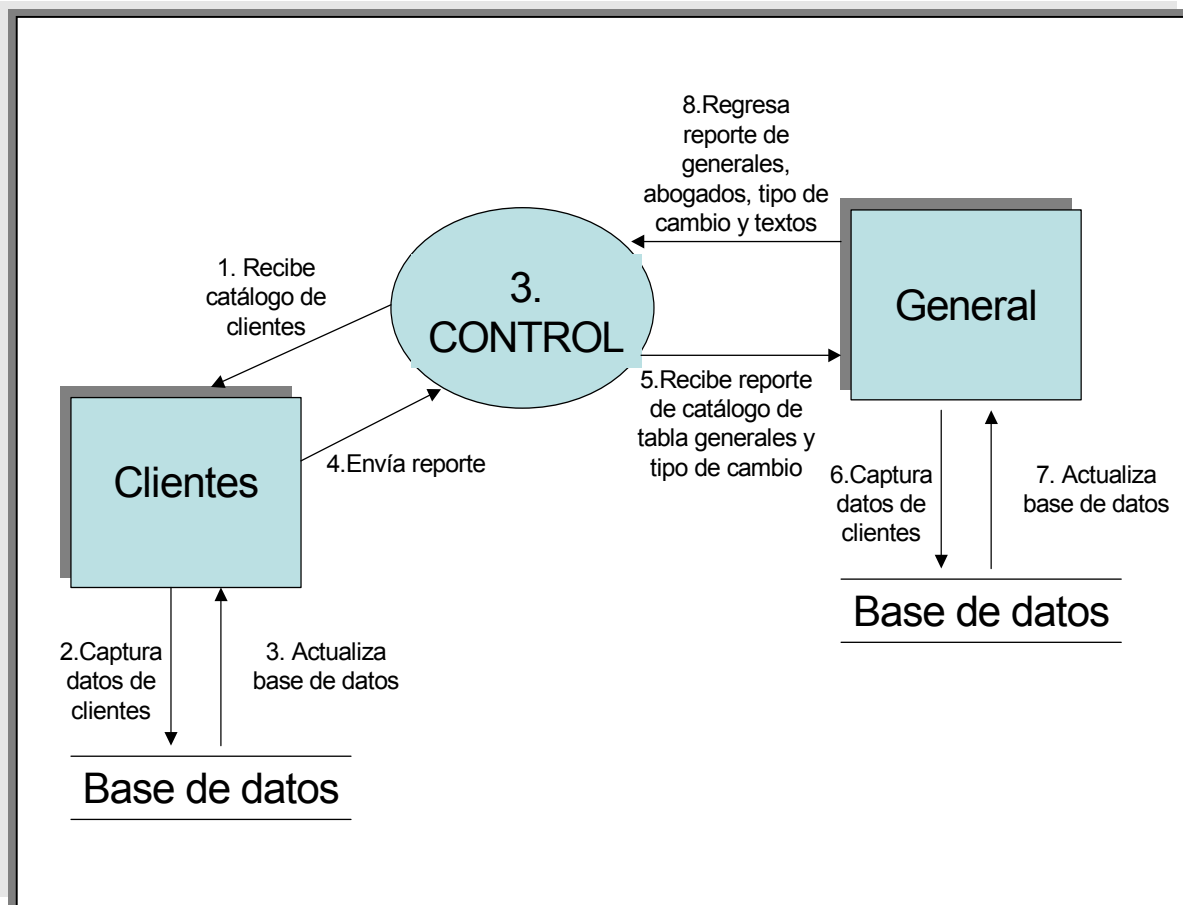
4.8.1 Diagrama de flujo de datos general del Sistema Jurídico de Cuentas por Cobrar (SJCC).



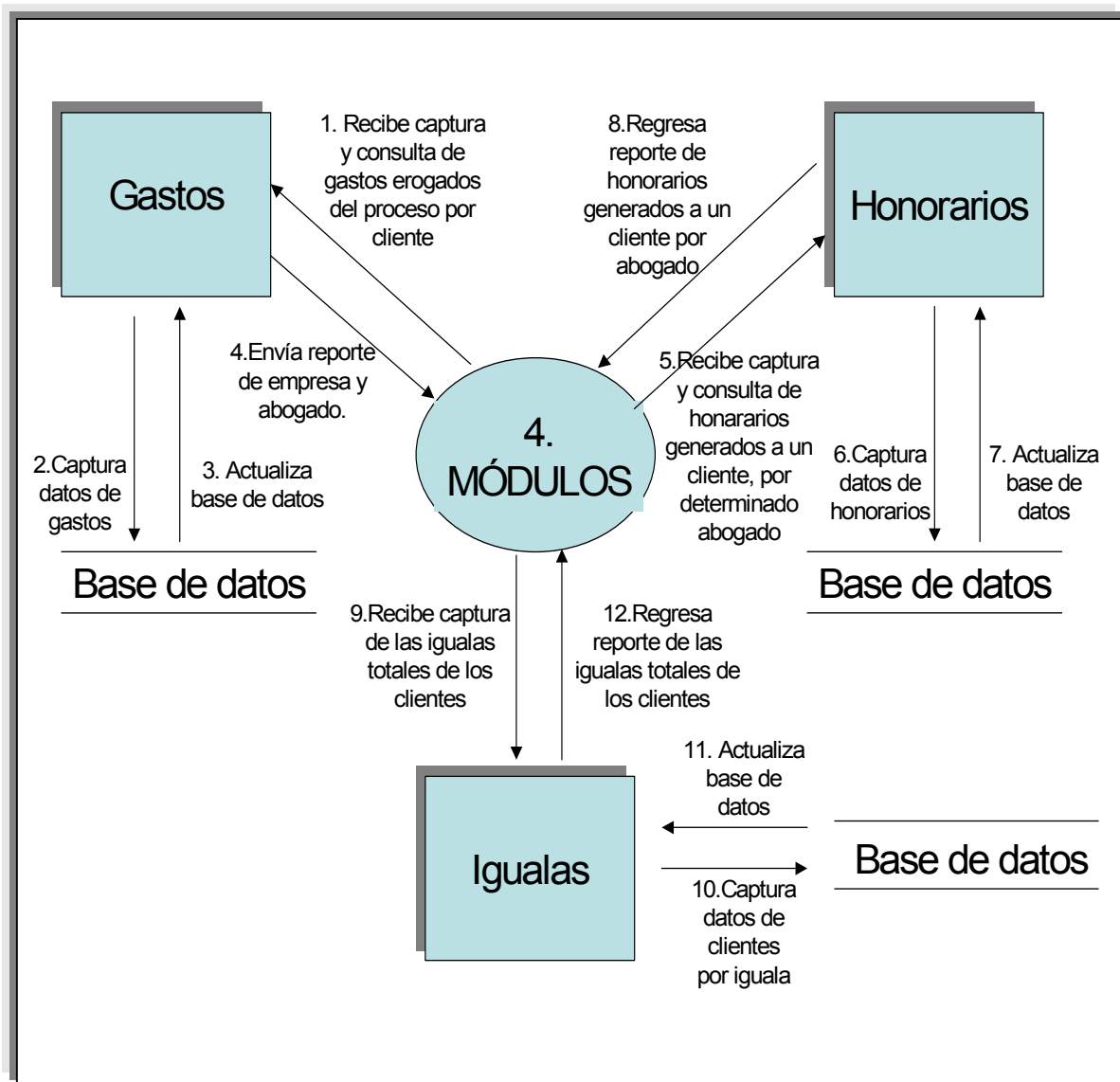
4.8.2 Diagrama de flujo de datos del proceso 2 (Archivo).



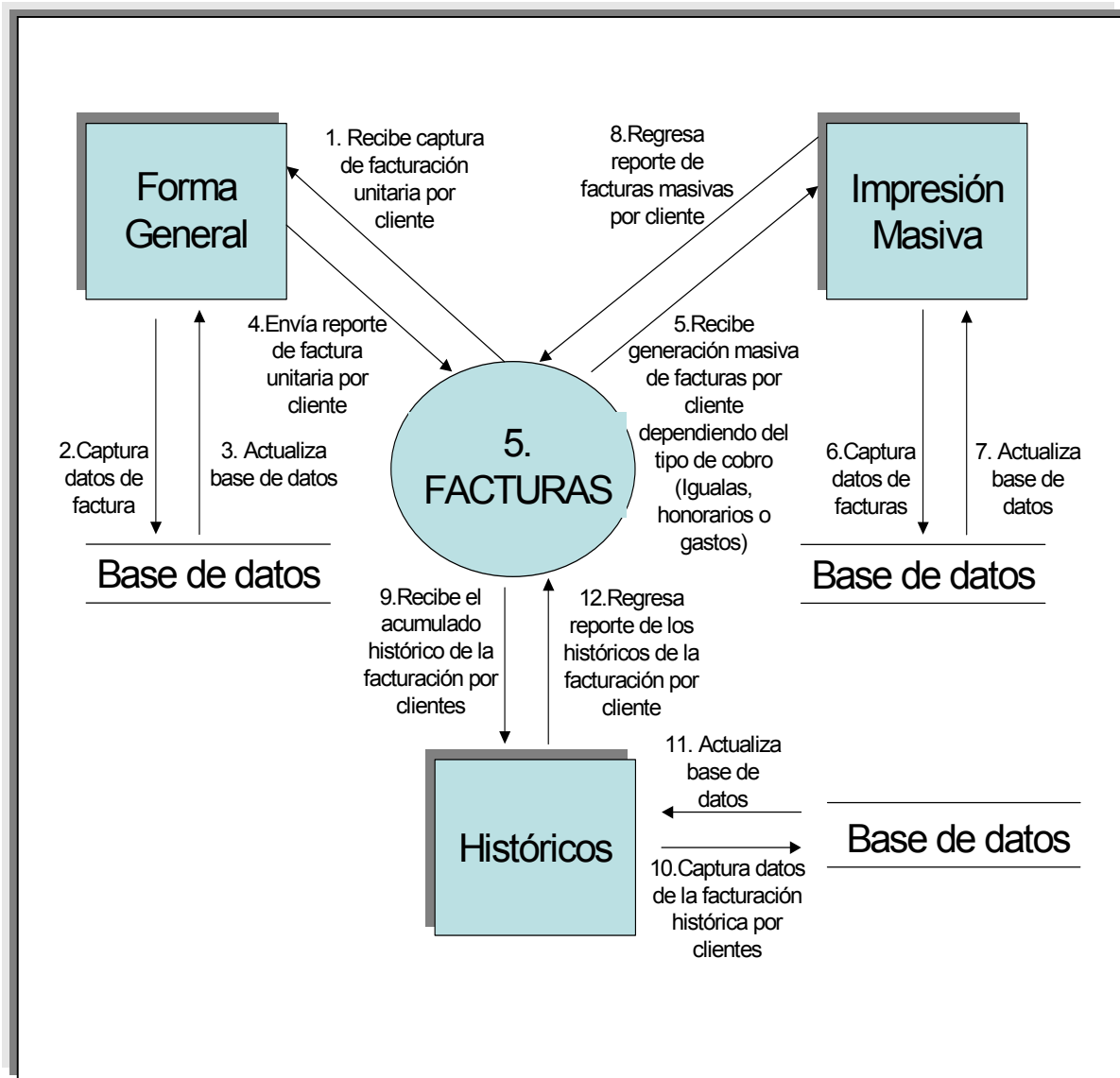
4.8.3 Diagrama de flujo de datos del proceso 3 (Control).



4.8.4 Diagrama de flujo de datos del proceso 4 (Módulos).



4.8.5 Diagrama de flujo de datos del proceso 5 (Facturas).



5. DISEÑO

El diseño del software lo podemos contemplar como la primera opción de tres actividades técnicas (*diseño, codificación y prueba*). De esta manera podemos decir que cada actividad transforma la información requerida de manera que finalmente se produce un software para computadora útil.

Para poder diseñar tenemos que tomar en cuenta los requerimientos del programa, establecidos mediante los modelos de información funcional y de comportamiento.

5.1 Conceptos para el diseño de software

Se establece entonces por medio de alguna de las metodologías de diseño, el diseño de datos, el diseño arquitectónico y el diseño procedimental. El diseño de datos transforma el modelo del campo de información, creado durante el análisis, en las estructuras de datos que se van a requerir para implementar el software. El diseño arquitectónico define las relaciones entre los principales elementos estructurados del sistema. El diseño procedimental transforma los elementos estructurales en una descripción procedimental del software. Se genera el código fuente y, para integrar y validar el software, se llevan a cabo las pruebas.

En el costo de la ingeniería del software las fases de diseño, codificación y prueba del sistema absorben el 75% o más del mismo. El diseño es el proceso en el que se asienta la calidad del desarrollo del software. Los requerimientos del cliente se pueden traducir de una única manera en el diseño del sistema para tener un producto o sistema final. El diseño del software sirve como base de todas las posteriores etapas del desarrollo y de la fase de mantenimiento.

El diseño del software es un proceso mediante el cual se traducen los requisitos en una representación del software. Para el punto de vista de la gestión del proyecto, podemos tener dos pasos para el diseño del software. El diseño preliminar, que es aquel que se encarga de transformar los requerimientos en los datos y la arquitectura del software. El

diseño detallado se ocupa del refinamiento de la representación arquitectónica que lleva a una estructura de datos detallada y a las representaciones algorítmicas del software. Además del diseño preliminar y detallado, muchas aplicaciones modernas requieren una actividad distinta de diseño de la interfaz. El diseño de la interfaz establece la disposición y los mecanismos para la interacción hombre-máquina.

Diseño y calidad del software.

Se establecen unos criterios para evaluar la calidad de una representación del diseño. Los cuales se muestran a continuación:

- Debe existir una organización jerárquica en un diseño y que haga un uso inteligente de control entre los componentes del software.
- Un diseño debe ser modular, el software se debe dividir de manera lógica en elementos que realicen funciones y subfunciones específicas.
- Un diseño debe contener representaciones distintas y separadas de los datos y de los procedimientos.
- Un diseño debe tener una secuencia a módulos (p. Ej.: subrutinas o procedimientos) que exhiban características funcionales independientes.
- Un diseño debe llevar a interfaces que reduzcan la complejidad de las conexiones entre los módulos y el entorno exterior.
- Un diseño debe obtenerse mediante un método que sea reproducible y que esté establecido por la información obtenida durante el análisis de los requerimientos del software.

Fundamentos del diseño.

Abstracción.

Pueden formularse muchos niveles de abstracción cuando se considera una solución modular para cualquier problema. Para el nivel superior de la abstracción se considera una solución en términos amplios usando el lenguaje del entorno del problema. En cambio para los niveles inferiores de abstracción se toma una orientación más procedimental. Y para el nivel más bajo de abstracción, se establece la solución de forma que pueda implementarse directamente.

Cada paso de un proceso de ingeniería de software es un refinamiento del nivel de abstracción de la solución de software. Durante la ingeniería del sistema, el software se considera como un elemento de un sistema basado en computadora.

De acuerdo como nos desplazamos por los distintos niveles de abstracción, trabajamos para definir abstracciones de datos y de procedimientos. Una abstracción procedimental es una determinada secuencia de instrucciones que tienen una función limitada y específica. Un ejemplo de abstracción procedimental podría ser la palabra “detenerse” en un cruce de autos. Una abstracción de datos es una determinada colección de datos que describen un objeto. Un ejemplo de esto podría ser un “factura_contabilidad”.

Por último tenemos la forma de abstracción que se utiliza en el diseño del software que es la abstracción de control. Al igual que la abstracción procedimental y de datos, la abstracción de control implica un mecanismo de control de programa, sin especificar los detalles internos.

Refinamiento.

El refinamiento sucesivo es una primera estrategia de diseño descendente propuesta por Niklaus Wirth. El refinamiento es realmente un proceso de elaboración. Se empieza declarando la función (o una descripción de la información) definida a un nivel superior de abstracción. La declaración describe la función o la información conceptualmente, pero no proporciona información sobre el funcionamiento interno de la función o sobre la estructura

interna de la información. El refinamiento hace que el diseñador amplíe la declaración original, dando cada vez más detalles conforme se produzcan los sucesivos refinamientos (elaboraciones).

Modularidad.

Un módulo es aquel que ejecuta independientemente una función específica, el cual es un subprograma de tamaño pequeño o modesto, por lo tanto si se elimina de un sistema, únicamente deshabilita la función que este desempeña en él. De igual manera, si se sustituye este módulo por otra versión del mismo, el programa se ejecutará con un cambio pequeño o casi nulo. Si basamos el desarrollo del software bajo este principio se dice que el diseño es modular.

Para poder hacer un diseño modular en primer lugar se deben seleccionar módulos como funciones independientes y posteriormente, tratar de subdividir cualquiera que sea muy grande. Si se presenta módulos muy pequeños, debemos evitar el intentar incluir un módulo pequeño en otro más grande. Por otra parte si se tiene un módulo grande pero que es bastante claro y simple y representa una sola función, entonces no tendría sentido el intentar subdividirlo.

Cohesión.

La cohesión es una extensión natural del concepto de ocultación de información, la cual sugiere que los módulos se deberían especificar y diseñar para que la información (procedimientos y datos) contenida dentro de un módulo sea inaccesible a otros módulos que no necesiten esa información, entonces un módulo con cohesión realiza una sola tarea dentro de un procedimiento de software, requiriendo poca interacción con los procedimientos que se realizan en otras partes del programa. Dicho de manera sencilla, un módulo con cohesión debería (idealmente) hacer una sola cosa.

Acoplamiento

El acoplamiento es una medida de la interconexión entre los módulos de una estructura de programa. Al igual que la cohesión, el acoplamiento puede representarse en un espectro. El acoplamiento depende de la complejidad de la interfaz entre los módulos, el punto en el que se entra o se hace referencia al módulo y qué datos pasan a través de la interfaz.

En el diseño del software, intentamos conseguir el menor nivel posible de acoplamiento. Las conexiones sencillas entre los módulos hacen que el software sea más fácil de entender y menos dado al “efecto ola” causado cuando ocurren errores en un lugar y se propagan a través del sistema.

5.2 Diseño del Sistema

A continuación se presenta de acuerdo a la propuesta de desarrollo el diseño del sistema, el diagrama de flujo de datos general del SJCC.

En esta sección se presenta el diagrama jerárquico funcional del SJCC y el desglose de todos los módulos que contiene el sistema.

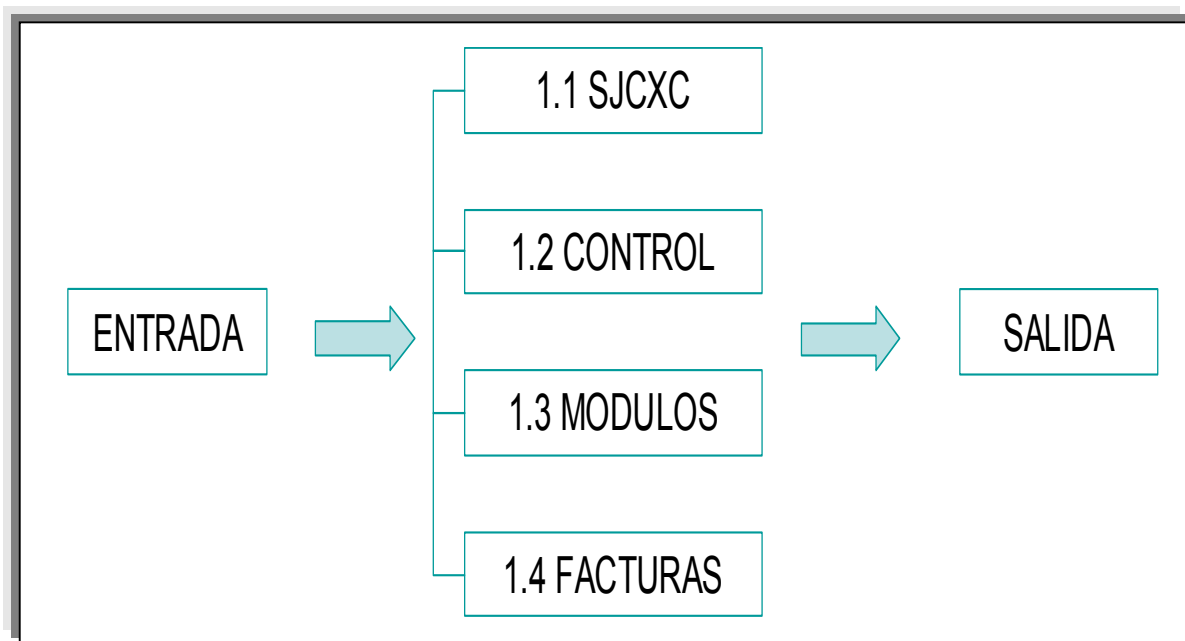


Fig. 5.2.1 Diagrama de Flujo de Datos General del SJCC

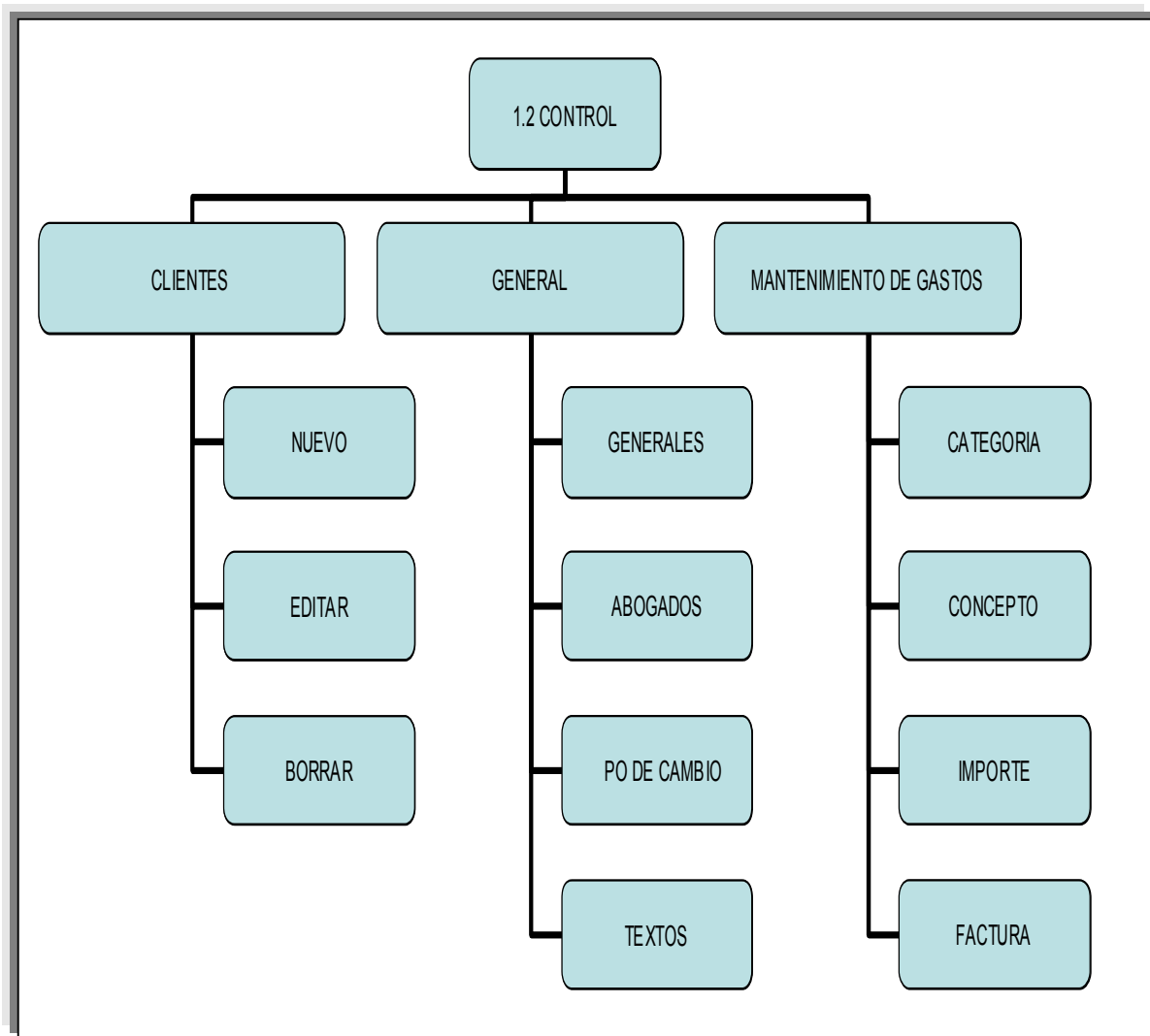


Fig. 5.2.2 Diagrama Jerárquico Funcional de Control

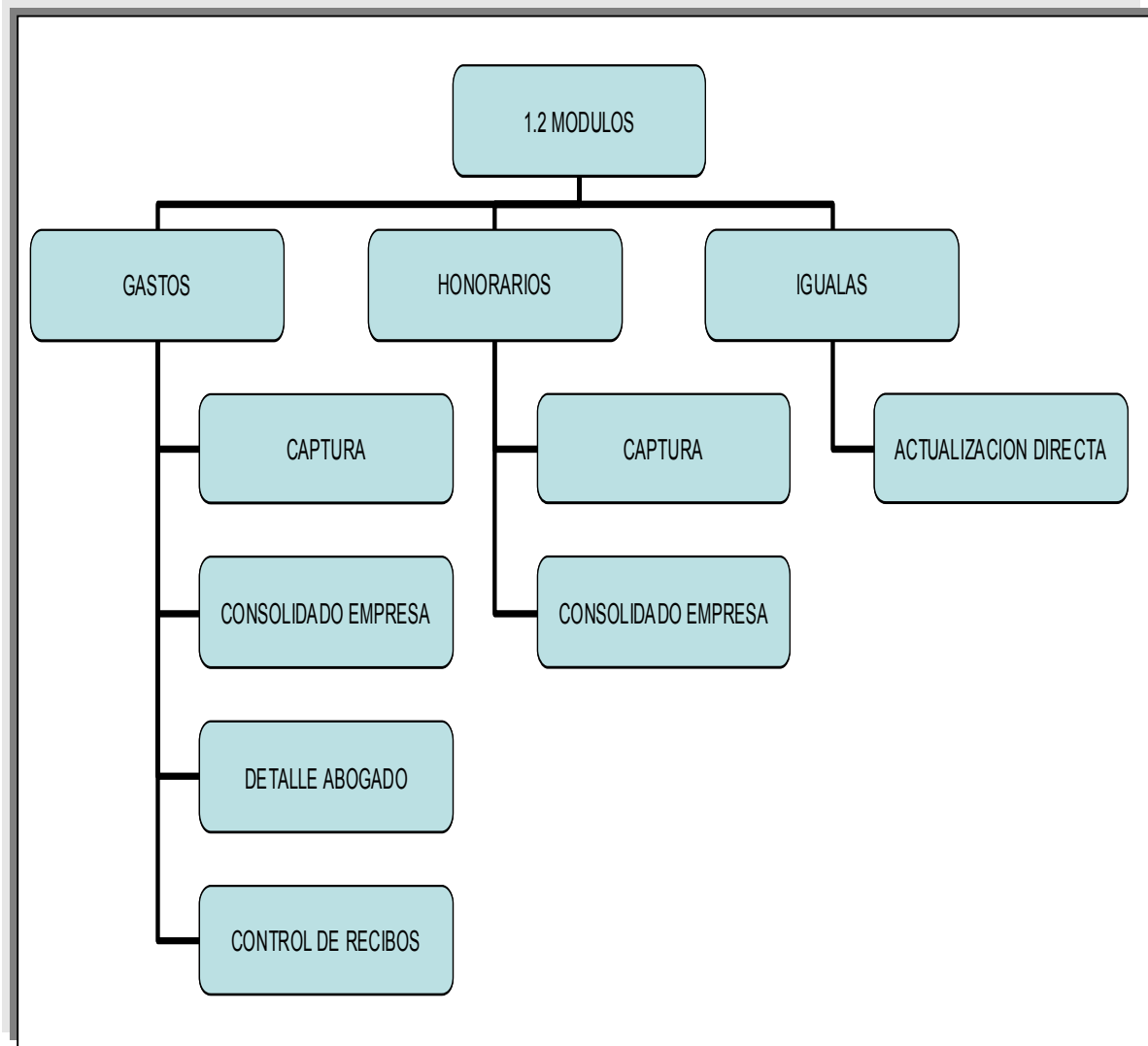


Fig. 5.2.3 Diagrama Jerárquico Funcional de Módulos

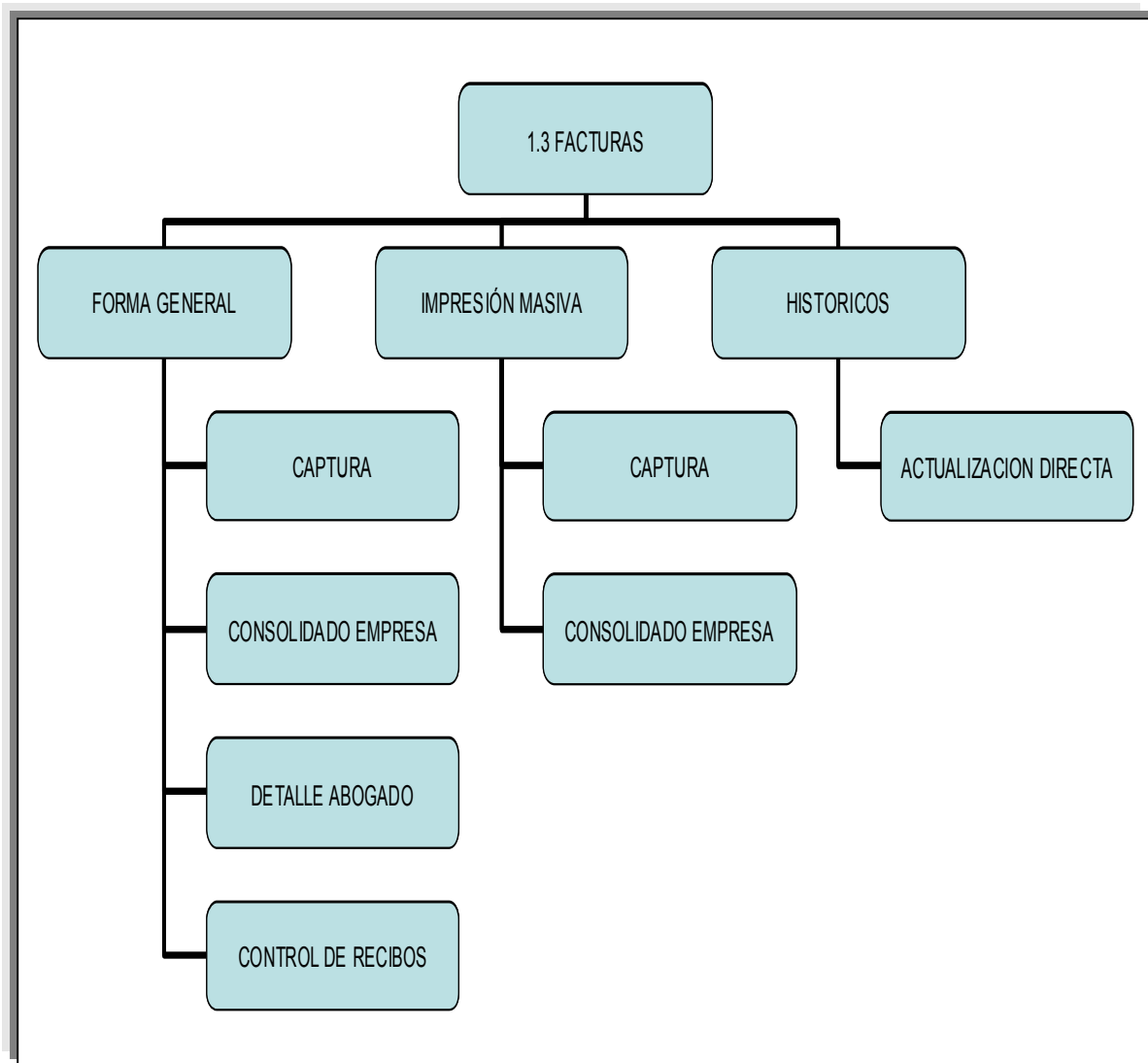


Fig. 5.2.4 Diagrama Jerárquico Funcional de Facturas

5.3 Diccionario de Datos

Los diccionarios de datos son el segundo componente del análisis del flujo de datos, después del Diagrama Entidad-Relación. En sí mismos los diagramas de flujo de datos no describen por completo el objeto de la investigación. El diccionario de datos proporciona información adicional sobre el sistema.

Un diccionario de datos es una lista de todos los elementos incluidos en el conjunto de los diagramas de flujo de datos que describen un sistema. Los elementos principales en un sistema, son el flujo de datos, el almacenamiento de datos y los procesos. El diccionario de datos almacena detalles y descripciones de estos elementos.

Si los analistas desean conocer cuántos caracteres hay en un dato, con qué otros nombres se le conoce en el sistema, o en donde se utilizan dentro del sistema deben ser capaces de encontrar las respuestas en un diccionario de datos desarrollado apropiadamente.

El diccionario de datos se desarrolla durante el análisis de flujo de datos y ayuda al analista involucrado en la determinación de los requerimientos del sistema, también el contenido del diccionario de datos se utiliza durante el diseño del sistema.

En informática, el diccionario de datos es la base de datos acerca de la terminología que se utilizará en un sistema de información. Para comprender mejor el significado de un diccionario de datos, puede considerarse su contenido como "datos acerca de los datos"; es decir, descripciones de todos los demás objetos (archivos, programas, informes, sinónimos, etcétera) existentes en el sistema. Un diccionario de datos almacena la totalidad de los diversos esquemas y especificaciones de archivos, así como sus ubicaciones. Si es completo incluye también información acerca de qué programas utilizan qué datos, y qué usuarios están interesados en unos u otros informes. Por lo general, el diccionario de datos está integrado en el sistema de información que describe.

El diccionario de datos define los datos haciendo lo siguiente:

- Describe el significado de los flujos y almacenes que se muestran en el Diagrama de Flujo de Datos.

- Describe la composición de agregados de paquetes de datos que se mueven a lo largo de los flujos, es decir, paquetes complejos que pueden descomponerse en unidades más elementales.
- Describe la composición de los paquetes de datos de los almacenes.
- Especifica los valores y unidades relevantes de piezas elementales de información en los flujos de datos y los almacenes de datos.

Notación del diccionario de datos.

- = está compuesto de
- + y
- () optativo (puede estar presente o ausente)
- {} iteración
- [] seleccionar una de varias alternativas
- ** comentario
- @ identificador (campo clave) para un almacén
- | separa opciones alternativas en la construcción

A continuación se presenta el diccionario de datos ocupado para el diseño del sistema.

Tabla Detalle Gastos

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
tcli_num	Número de Cliente	Numérico	4	Foránea
rec_gtos	Número de recibo de gastos	Numérico	6	Primaria
gtos_fech	Fecha en la que se incurrió el gasto	Fecha	8	Foránea
gtos_atn	Abogado que atendió o genero el gasto	Carácter	20	-
gtos_impab	Importe del gasto a cubrir al abogado	Moneda	8	-
gtos_impem	Importe del gasto a cobrar a la empresa	Moneda	8	-
gtos_con	Concepto del gasto	Memo	4	-
gtos_aut	Nombre de quien autoriza el gasto	Carácter	20	-
gtos_asun	Asunto o expediente relacionado al gasto generado	Carácter	75	-
gtos_abopa	Indicador de pago realizado al abogado de este gasto	Numérico	1	-

Tabla de Clientes

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
tcli_num	Número de Cliente	Numérico	4	Primaria
tcli_rfc	R.F.C.	Carácter	22	-
tcli_cia	Razón Social de la Compañía	Carácter	80	-
tcli_dir1	Calle y número	Carácter	50	-
tcli_dir2	Colonia y Delegación o Municipio	Carácter	50	-
tcli_cd	Ciudad	Carácter	25	-
tcli_edo	Estado	Carácter	19	-
tcli_cp	Código Postal	Carácter	8	-
tcli_tel1	Teléfono de la organización 1	Numérico	8	-
tcli_tel2	Teléfono de la organización 2	Numérico	8	-
tcli_fax	Número telefónico de Fax	Numérico	8	-
tcli_email	Dirección de correo electrónico de Internet	Carácter	50	-
tcli_obs	Observaciones asociadas a este cliente	Memo	4	-
tcli_mon	Moneda en la que se le factura al cliente	Carácter	5	-
tcli_prov	Número asignado por el cliente que nos identifica como proveedores del mismo	Carácter	10	-

Tabla de Detalle Facturas

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
Rcc_fac	Número de la Factura	Numérico	6	Primaria
det_desc	Concepto de la factura	Memo	4	-
det_impgral	Importe de la factura	Moneda	8	Foránea
det_cat	Categoría de la factura	Carácter	10	-

Tabla Factura

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
rcc_fac	Número de Factura	Numérico	6	Primaria
tcli_num	Número de cliente	Numérico	4	Foránea
rcc_imp	Importe de la factura	Numérico	10	-
rcc_implt	Importe en letra de la factura	Memo	4	-
rcc_fecha	Fecha de emisión de la factura	Fecha	8	-
rcc_mes	Mes al que corresponde la factura por concepto de iguala	Carácter	12	-
rcc_fechp	Fecha estimada para pago de la factura	Fecha	8	-
rcc_sta	Estatus de pago de la factura: "Pagado, cancelado, pendiente, jurídico y otros"	Carácter	10	-
rcc_obs	Observaciones o notas importantes de la factura	Memo	4	-

Tabla Honorarios

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
tcli_num	Número de cliente	Numérico	4	Primaria
rcc_fac	Número de factura	Numérico	6	Foránea
hon_fech	Fecha de generación de honorarios	Fecha	8	Foránea
hon_atn	Personal que atendió la consulta	Carácter	20	-
hon_imp	Importe generado de la consulta	Moneda	8	-
hon_con	Descripción de la consulta solicitada	Memo	4	-
hon_sol	Solicitante de la consulta	Carácter	50	-
hon_tpo	Tiempo de duración de la consulta	Carácter	6	-

Tabla General

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
mes	Campo de los meses del año y períodos de consolidación	Carácter	15	-
bancos	Nombre de bancos e instituciones financieras	Carácter	20	-
estatus	Estatus general	Carácter	15	-
tipodepago	Tipo de pago general	Carácter	15	-
varios_val	Valores numéricos utilizados para cálculos: IVA, ISR, porcentaje de ganancias en gastos, etc.	Numérico	6	-
varios_txt	Texto del concepto insertado en las facturas generadas automáticamente en forma masiva	Carácter	25	-
abogados	Abogados o personal legal de la firma o despacho	Carácter	20	-
band_abog	Rango de los abogados	Numérico	1	-
tarifa	Tarifa de cada abogado	Carácter	8	-
textos	Texto por concepto de asesorías	Memo	4	-
categoría	Categoría de los pagos	Carácter	10	-

Tabla Iguales

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
tcli_num	Número de cliente	Numérico	10	Primaria
igu_imp	Importe mensual de la iguala	Moneda	8	-

Tabla Tarifa

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
tcli_num	Número de cliente	Numérico	4	Primaria
tar_soc	Tarifa del socio del despacho	Moneda	8	-
tar_aso	Tarifa del asociado	Moneda	8	-
tar_pas	Tarifa del pasante del despacho	Moneda	8	-

Tabla Recibos

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
tcli_num	Número del cliente	Numérico	4	-
tcli_rfc	RFC	Carácter	22	-
tcli_cia	Razón Social de la compañía	Carácter	50	-
tcli_dir1	Calle y número	Carácter	50	-
tcli_dir2	Colonia o Delegación o Municipio	Carácter	50	-
tcli_cd	Ciudad	Carácter	25	-
tcli_edo	Estado	Carácter	19	-
tcli_cp	Código Postal	Numérico	8	-
rcc_ser	Número de serie de la factura	Carácter	1	-
rcc_fac	Número de factura	Numérico	5	Foránea
rcc_imp	Importe numérico de la factura	Moneda	8	-
rcc_implt	Importe en letra de la factura	Memo	4	-
rcc_fecha	Fecha de emisión de la factura	Fecha	8	-
rcc_fechp	Fecha de pago de la factura	Fecha	8	-
rcc_tp	Tipo de Pago	Carácter	10	-
rcc_obs	Observaciones asociadas al cliente	Memo	4	-
rcc_concep	Concepto del recibo emitido	Memo	4	-
rcc_mes	Mes al que corresponde el recibo	Carácter	12	-
rcc_sta	Estatus del recibo	Carácter	15	Foránea
rcc_numche	Número de folio del cheque de pago	Carácter	15	-
rcc_banc	Banco o institución del documento	Carácter	20	-
tcli_mon	Tipo de moneda	Carácter	5	-

Tabla Recibos Detalle Factura

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
rcc_fac	Número de factura	Numérico	6	Primaria
det_desc	Descripción de concepto de la factura	Carácter	254	-
det_impgral	Importe general de la factura	Moneda	8	-
det_impptos	Importe de gastos generales	Moneda	8	-

Tabla Desglose de Gastos

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
rec_desc	Concepto del recibo general de gastos	Carácter	30	Primaria
rec_gtos	Número de recibo	Numérico	6	Candidato
tcli_num	Número de cliente	Numérico	4	-
rec_imp	Importe del recibo	Numérico	10	-
rec_implt	Cantidad en letra del importe	Memo	4	-
rec_fecha	Fecha de emisión del recibo	Fecha	8	-
rec_fechp	Fecha de pago del recibo	Fecha	8	-
rec_sta	Estatus del recibo	Carácter	10	-
rec_obs	Observaciones del cliente del recibo	Memo	4	-

Tabla Tipo de Cambio

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
tc_fecha	Fecha del pago	Fecha	8	Primaria
tc_importe	Importe del pago	Moneda	8	-

Tabla Pagos

NOMBRE	DESCRIPCION	TIPO CARÁCTER	LONGITUD	LLAVE
tcli_num	Número de cliente	Numérico	4	Primaria
pag_imp	Importe del pago del cliente	Numérico	10	-
pag_fechp	Fecha de pago del cliente	Fecha	8	Foránea
pag_tp	Tipo de pago	Carácter	10	-
pag_numche	Número de documento generado por el pago	Carácter	15	-
pag_banc	Banco o institución del documento	Carácter	12	-
pag_obs	Observaciones o notas importantes de los pagos	Memo	4	-

En la siguiente figura se hace referencia al diagrama Entidad-Relación.

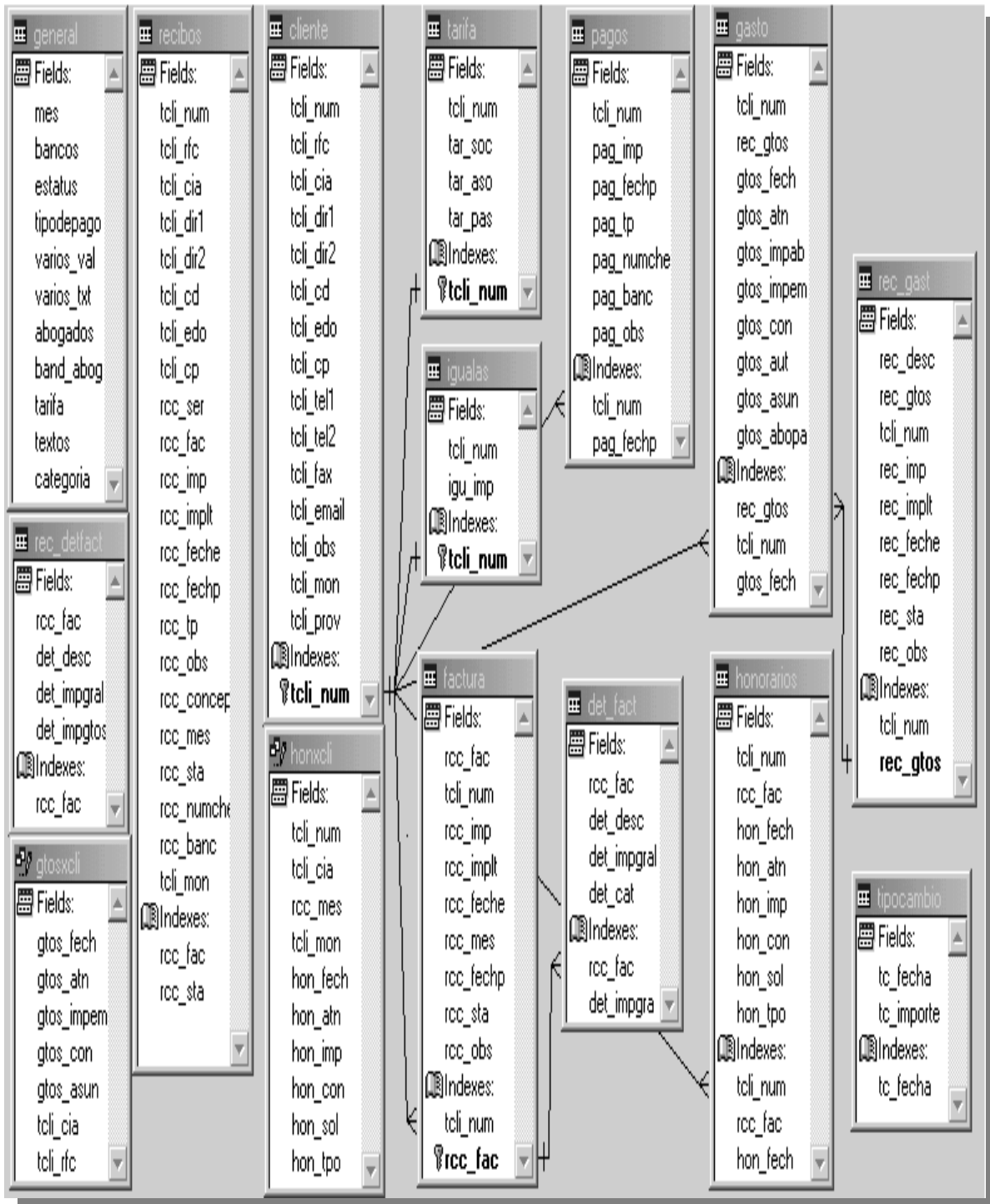


Fig. 5.3.1 Diagrama Entidad-Relación

6. DESARROLLO

En esta etapa se pretende desarrollar el sistema a través de los pasos de la Ingeniería del Software que se han tratado hasta ahora, con un objetivo específico: traducir las representaciones del software a una forma que pueda ser comprendida por la computadora, este proceso de transformación de las especificaciones que se obtuvieron en la fase de diseño es conocida como implementación y desarrollo del sistema.

6.1 Selección del software

La selección del software depende de la función que deba desarrollar, el tiempo que requiera el sistema para programarse y también el costo que involucra estos factores, así como las necesidades del usuario, aparte que se escoge una herramienta que se pueda ejecutar en el sistema operativo Windows.

Para escoger el software de programación se tomo en cuenta lo siguiente:

- *Conocimiento del lenguaje de programación.* Aunque puede resultar sencillo para los programadores experimentados aprender un nuevo lenguaje, se necesita cierta práctica antes de adquirir cierta competencia, entonces es preferible elegir un lenguaje con el que se esté familiarizado.
- *Proyectos previos desarrollados en esté lenguaje.* Esto se relaciona mucho con el punto anterior, ya que si se ha trabajado en proyectos previos con algún tipo de lenguaje se esta familiarizado con este.

Windows

Es un entorno operativo para gráficos elaborado por la compañía Microsoft que es integrado con MS-Dos. Proporciona un entorno de trabajo parecido al de Macintosh en el cual cada aplicación activa se visualiza en una ventana movable y que es redimensionable sobre la pantalla. Windows no sólo es un software que se ejecuta sobre el sistema operativo, más bien en cierta medida lo reemplaza, proporcionando al usuario un entorno gráfico amigable para su interacción con el hardware y el software.

El entorno gráfico de Windows permite que las aplicaciones sean más sencillas de aprender y de utilizar, ya que en lugar de aprender a escribir largos comandos, el usuario simplemente selecciona una opción o comando de un menú con un “click” del ratón. También aparecen cuadros de diálogo cuando un programa necesita información o cuando el usuario necesita tomar alguna decisión sobre los programas.

Además, las aplicaciones para Windows tienen generalmente una interfaz de usuario consistente, es decir que los usuarios pueden disponer de más tiempo para dominar la aplicación sin necesidad de preocuparse de qué teclas deben oprimirse, dentro de los menús o cuadros de diálogo.

Lenguajes de Programación

Es necesario establecer que debido a los avances tecnológicos se establece un análisis del software, para definir qué herramienta de programación es la más adecuada para definir una interfaz gráfica hacia el usuario, estos son algunos de los lenguajes de programación que se conocen para este tipo de programación y de los más actuales:

- Visual Basic
- Delphi
- Visual FoxPro

Visual Basic

Visual Basic, es una de las herramientas más productivas que se ocupan para el desarrollo de sistemas, esto por las creaciones rápidas y soluciones a negocios bajo una plataforma Windows y para Web.

Para escribir programas en Windows, la programación orientada a eventos es conveniente, ya que es un método flexible. A Visual Basic no se le puede definir como un sistema de programación orientada, ya que solamente se crean objetos llamados formas y controles, que permiten que la aplicación funcione, y ya que estos objetos carecen de las propiedades de herencia y polimorfismo, las cuales tiene que estar presentes en un

verdadero ambiente orientado a objetos. Una de las ventajas de ocupar Visual Basic es que proporciona una interfaz consistente y manejable para desplegar la información en muchas aplicaciones distintas que comparten controles similares.

Al aplicar Visual Basic se reduce considerablemente las dificultades y la cantidad de tiempo de programación en Windows, esto por que es un lenguaje de interpretación más que de compilación, es decir las aplicaciones están listas para correr tan pronto como se termine de escribir el código. Visual Basic es un lenguaje de programación que permite desarrollar aplicaciones y ventanas GUI (Graphic User Interface), que tiene un parecido similar al ambiente Windows y son muy familiares para el usuario.

Para poder instalar el programa de Visual Basic, es necesario asegurarse de que el equipo de cómputo cumpla con los requerimientos básicos:

- Procesador Pentium II a 650 Mhz o superior (recomendado).
- 32MB de memoria RAM.
- Espacio en disco duro de mínima 116MB, instalación máxima 135MB
- Drive CD-ROM .
- Monitor VGA, Súper VGA.
- Sistema Operativo Microsoft Windows 95 o superior

Delphi

Delphi Interprise una de las herramientas de desarrollo mas conocidas, desarrollado por la empresa Borland, se basa en la utilización de componentes o controles prefabricados para la creación de aplicaciones, personalizados mediante propiedades y que están dispuestos en ventanas.

Es un entorno de desarrollo simple, potente y flexible, cuenta con un sinfín de componentes, que facilitan de manera notable la creación de cualquier aplicación, utilizando componentes como:

- El formulario
- Múltiples ventanas
- El inspector de objetos
- Barra de botones
- Paleta de componentes
- El editor de código

Todo el desarrollo de aplicaciones en Delphi se encuentra íntimamente unido con la definición y el uso de objetos, que se describen por sus características y por su funcionamiento, lo cual permite utilizar y desarrollar componentes, que contarán con propiedades, eventos y métodos.

La programación en Delphi se realiza de manera modular, es decir, se cuenta con un módulo de inicio que se relaciona y depende de otros módulos para su funcionamiento.

Para poder manipular los datos se pueden ocupar herramientas tales como:

- Servidores de Internet
- Explorador de base de Datos
- SQL Builder
- InterBase
- Diseñador de Informes
- Clientes Internet
- Navegación y Edición de Datos
- Diseñador de módulos de datos

Otras características de Delphi son:

- Aplicación Multihilos
- Acceso a Bases de Datos
- Ensamblado Integrado
- Dispositivos Multimedia.

Visual FoxPro

Visual FoxPro es uno de los gestores de Bases de Datos relacionales más flexible y rápido que existen en el mercado, ya que dispone de un entorno de desarrollo complejo que esta orientado totalmente a la programación de objetos y que se integra altamente al sistema operativo Windows estándar, soporta interfaces Mdi, barras de herramientas, acceso a bibliotecas de enlace dinámico, ayuda sensible al contexto, fuentes múltiples.

También se puede acceder virtualmente a cualquier fuente de datos cuyo acceso se pueda hacer a través de ODBC (Open Database Connectivity). Este estándar, definido como Conectividad Abierta de Bases de Datos, permite abrir y consultar diferentes bases de datos, a través de un conjunto de controladores, manejando SQL como lenguaje de consulta.

Requerimiento Mínimo de Equipo

- Procesador Pentium II a 650 Mhz o superior (recomendado)
- Un PC compatible con Windows 95 o superior
- Monitor VGA o de mayor resolución.
- 32 MB en RAM
- Un Mouse.

Analizados los diferentes lenguajes de programación que pueden ser usados para la interfaz gráfica, se evaluaron algunas características propias de los mismos que pueden resultar de interés para la elección de la herramienta más conveniente para los fines que se buscan.

HERRAMIENTAS DE DISEÑO	EXCELENTE	BUENO	REGULAR
Visual Basic		X	
Delphi	X		
Visual FoxPro	X		

COSTOS Y DISPONIBILIDAD	EXCELENTE	BUENO	REGULAR
Visual Basic	X		
Delphi			X
Visual FoxPro	X		

ORIENTADA A OBJETOS	EXCELENTE	BUENO	REGULAR
Visual Basic			No Aplica
Delphi	X		
Visual FoxPro	X		

CONOCIMIENTO DEL LENGUAJE	EXCELENTE	BUENO	REGULAR
Visual Basic		X	
Delphi		X	
Visual FoxPro	X		

Con el análisis anterior, se ha concluido que la herramienta más adecuada para el proyecto es Visual FoxPro, ya que es una herramienta de programación muy poderosa para el manejo de componentes y su programación orientada a objetos. También porque se cuenta con la herramienta y se tiene mayor conocimiento de este lenguaje, además que es de fácil mantenimiento.

6.2 Etapa de desarrollo

En esta etapa se mencionan las características del lenguaje de programación, así como su entorno de desarrollo, con la finalidad de conocer más a fondo la herramienta que se utilizará para el desarrollo del sistema y que permita optimizar los recursos existentes.

Características de Visual FoxPro

Es un sistema de bases de datos relacionales que simplifica la administración de los datos y permite racionalizar el desarrollo de aplicación.

Con Visual FoxPro se permite la clasificación de los datos, definición de reglas para bases de datos y crear aplicaciones, facilita crear rápidamente formularios, informes y consultas a través de herramientas visuales de diseño y los asistentes, se pueden crear también rápidamente aplicaciones de presentaciones completas, esto porque proporciona un entorno integrado de desarrollo que contiene poderosos instrumentos de programación orientada a objetos, compatibilidad con OE y Actives, posibilidades cliente-servidor.

Con Visual FoxPro se da un gran salto para el desarrollo de aplicaciones xBase. Los métodos de acceso y asignación presentan una nueva dimensión para el control de datos que están colocados dentro de los objetos. La nueva galería de componentes y clases fundamentales facilita la transición hacia la creación de aplicaciones orientadas a objetos.

El soporte a COM permite una mejor y más fácil creación e implementación de los servidores COM. Se puede mencionar también:

- Posibilidad de compatibilidad con el lenguaje Dbase III y Dbase IV con el compilador de FoxPro.
- Sistema de programación multiusuario para uso en red, o de área local.
- Eficiencia en el manejo de ventanas de dialogo.
- Capacidad para dialogar con ventanas en su entorno Windows.

- Múltiples tablas de ayuda para la creación de programas con informes, etiquetas y herramientas de gran utilidad para el usuario.
- Campos memos para el almacenamiento de texto en las bases de datos.
- Generadores de informes y etiquetas.
- Manejador para convertir programas FoxPro (*.fxp) en ejecutables (ejecutable.exe).

Entorno de desarrollo de Visual FoxPro

La mayoría de la interfaz de desarrollo de Visual FoxPro está orientada a la creación rápida de aplicaciones (RAD). El principal objetivo es la creación de aplicaciones que sean seguras en un tiempo razonable, para realizar esto se deja que el sistema haga el trabajo a bajo nivel, dejando para el programador el trabajo visual y el análisis de la aplicación, entonces Visual FoxPro se apoya de las siguientes herramientas para lograr esto:

Un administrador de proyectos: se incorpora un poderoso gestor de proyectos que centraliza la gestión de todos los archivos que maneja la aplicación.

Un diseñador de formularios: totalmente enfocado a la creación de objetos, se pueden diseñar la forma gráfica final de una aplicación, la introducción de objetos de tipo estándar de Windows, como grupos de opciones, marcos de páginas, casillas de verificación.

Un diseñador de bases de datos: este gestor interactúa con el manejo de bases de datos que operan como referente de la información extra relacionada con los índices, relaciones, tablas, integridad referencial, reglas de validación, etcétera. De esta manera, en una base de datos se mantiene centralizada, unida y actualizada toda la información de funcionamiento de las tablas que están relacionadas con la base.

Diseñador de informes y etiquetas: se presenta al momento de visualizar los datos, en la impresora o en la pantalla.

Un asistente de consultas de vistas: si se requiere recuperar información estos generadores de consultas y vistas proporcionan una ayuda importante para la generación de expresiones SQL.

Un diseñador de menús: utilizado para elaborar menús acordes a las necesidades del usuario o la aplicación y poder sustituir el menú que tiene por definición Visual Fox Pro.

Asistentes: Esta herramienta permite crear nuevos tipos de ficheros de forma sencilla y rápida. A continuación se muestra el entorno de desarrollo de Visual Fox Pro.

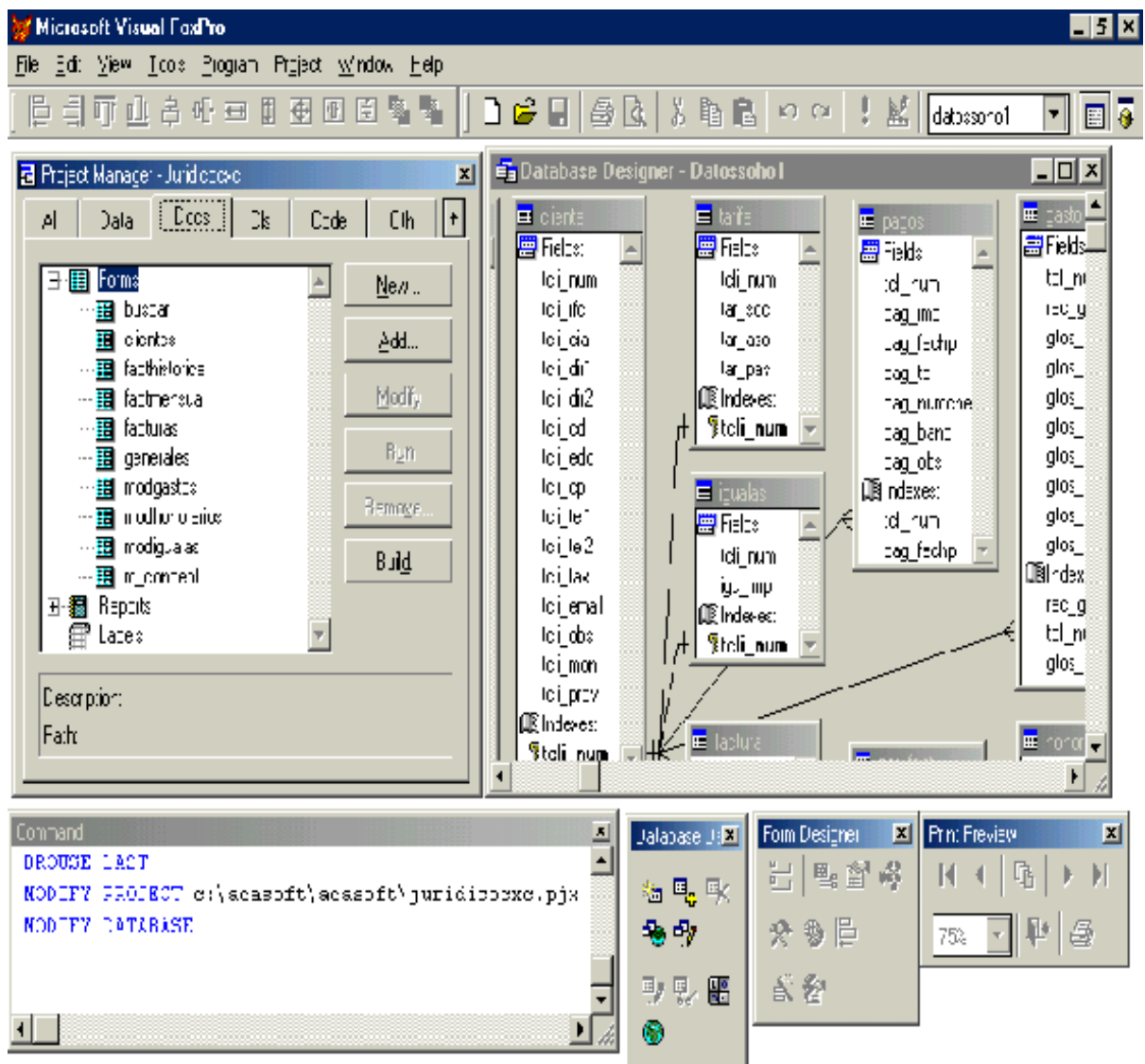


Fig. 6.2.1 Entorno de Desarrollo

6.3 Desarrollo del Sistema

Aquí se presenta el programa de desarrollo y una parte de la codificación del sistema, que permite apreciar más a fondo el crecimiento del proyecto, su funcionamiento e implementación.

Se realizó primero todos los formularios de la aplicación, a continuación se presenta un ejemplo de esto, el formulario *clientes*, el cual indica el catálogo de clientes existentes en el despacho, así como la información requerida de los mismos para la emisión de una factura, lo que muestra en la siguiente figura.

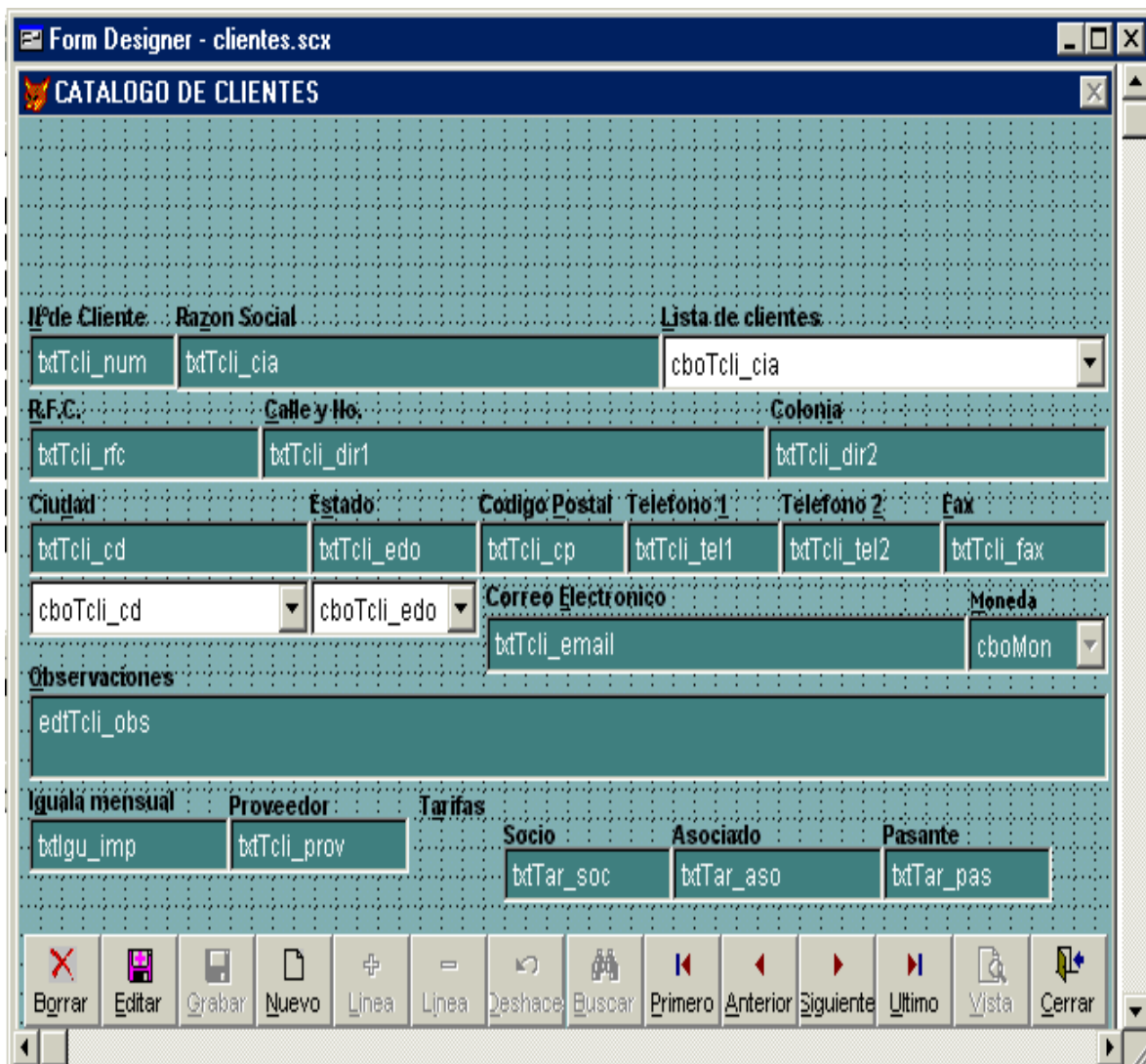
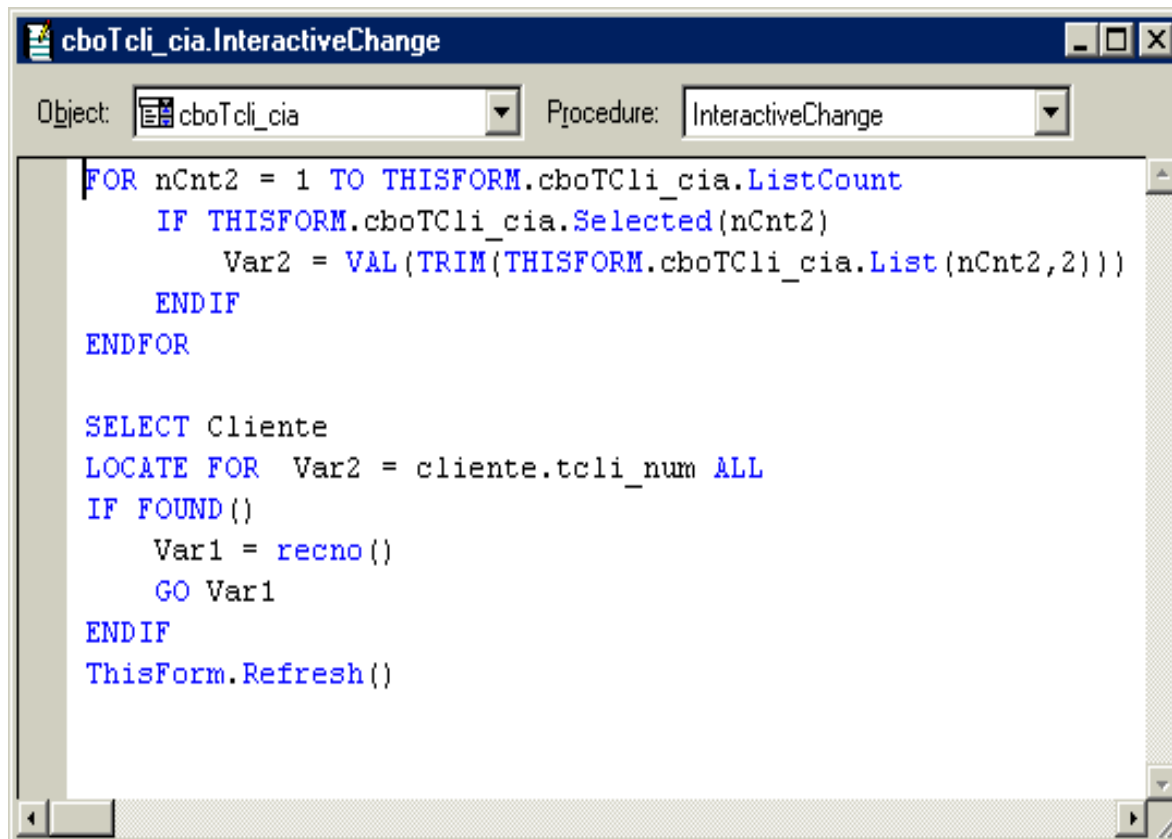


Fig. 6.3.1 Formulario de Catálogo de Clientes

Haciendo doble “click” sobre algunos de los objetos de Visual Fox Pro que aparecen en el formulario de *Cientes*, se puede escribir instrucciones que se ejecutarán cuando ocurra el evento asociado al control, para ejemplificar esto se aplica esto con el objeto *lista de clientes* (cboTcli_cia), que tiene asociado el siguiente código en el evento *InteractiveChange*, teniendo lo siguiente:



```
Object: cboTcli_cia Procedure: InteractiveChange

FOR nCnt2 = 1 TO THISFORM.cboTcli_cia.ListCount
    IF THISFORM.cboTcli_cia.Selected(nCnt2)
        Var2 = VAL(TRIM(THISFORM.cboTcli_cia.List(nCnt2,2)))
    ENDIF
ENDFOR

SELECT Cliente
LOCATE FOR Var2 = cliente.tcli_num ALL
IF FOUND()
    Var1 = recno()
    GO Var1
ENDIF
ThisForm.Refresh()
```

Fig. 6.3.2 Código en Visual Fox Pro

Cuando se ejecuta la aplicación se obtiene un formulario donde se despliega una lista con los nombres de los clientes del despacho ordenados alfabéticamente, cuando uno selecciona alguno se muestra el estatus que guarda el cliente respecto al despacho, número de cliente, razón social, teléfonos, correo electrónico, etcétera, así como los diferentes tipos de cargos de cobro que se le aplican dependiendo de las tarifas del abogado que lo atendió, todo esto con la finalidad de aplicar esta información en la factura que se le emitirá al cliente y tener guardada su información en la base de datos.

Esto se muestra a continuación en la figura 6.3.3

CATALOGO DE CLIENTES X

Rizo, Armida y Marroquin, S.C.

Nº de Cliente	Razon Social	Lista de clientes
185	ALTER OPERADORA DE TIENDAS, S.A. DE C.V.	ALTER OPERADORA DE TIENDAS, S.A. DE C.V.
R.F.C.	ALTER OPERADORA DE TIENDAS, S.A. DE C.V.	
AOT950706CD5	ANDRES MARCELO ARMIDA GRAHAM.	
Ciudad	AQUANOVA CORPORATIVO, S.A. DE C.V.	
México D.F.	ARBOMEX, S.A. DE C.V.	
	ARROMEX, S.A. DE C.V.	
	ASTRAZENECA, S.A. DE C.V.	
Observaciones	AUTOMOTRIZ LLANTERA, S.A. DE C.V.	

Iguala mensual	Proveedor	Tarifas		
\$0.00		Socio	Asociado	Pasante
		\$0.00	\$0.00	\$0.00

Borrar	Editar	Grabar	Nuevo	Línea +	Línea -	Deshacer	Buscar	Primero	Anterior	Siguiete	Ultimo	Vista	Cerrar

Fig. 6.3.3 Listado de clientes en el formulario Clientes

Vamos a mostrar un segundo ejemplo con otro formulario del sistema, en esta ocasión se muestra el formulario de *facturación unitaria*, en el cual se efectúan modificaciones de forma manual a las facturas emitidas o que necesitan alguna corrección, también muestra la factura de algún cliente para su ajuste o reimpresión. Como se muestra en la siguiente figura.

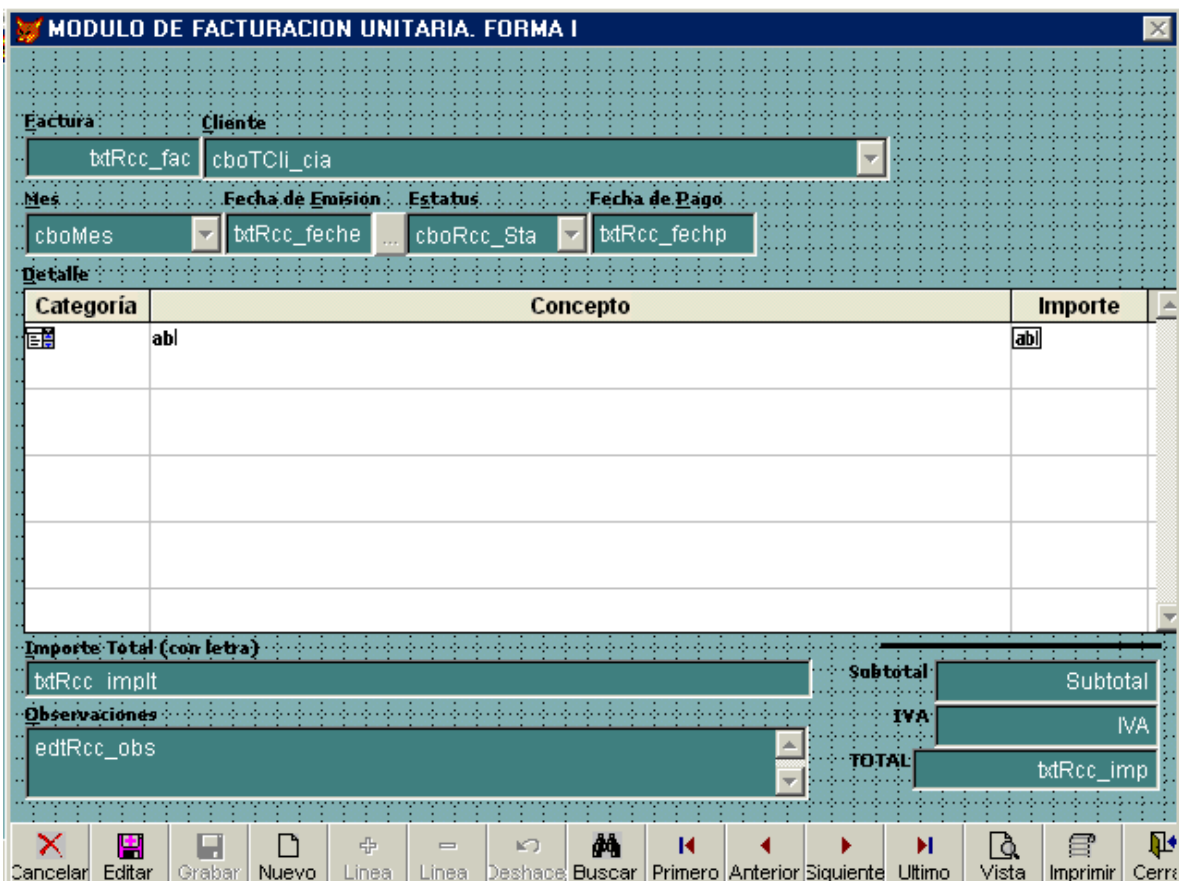
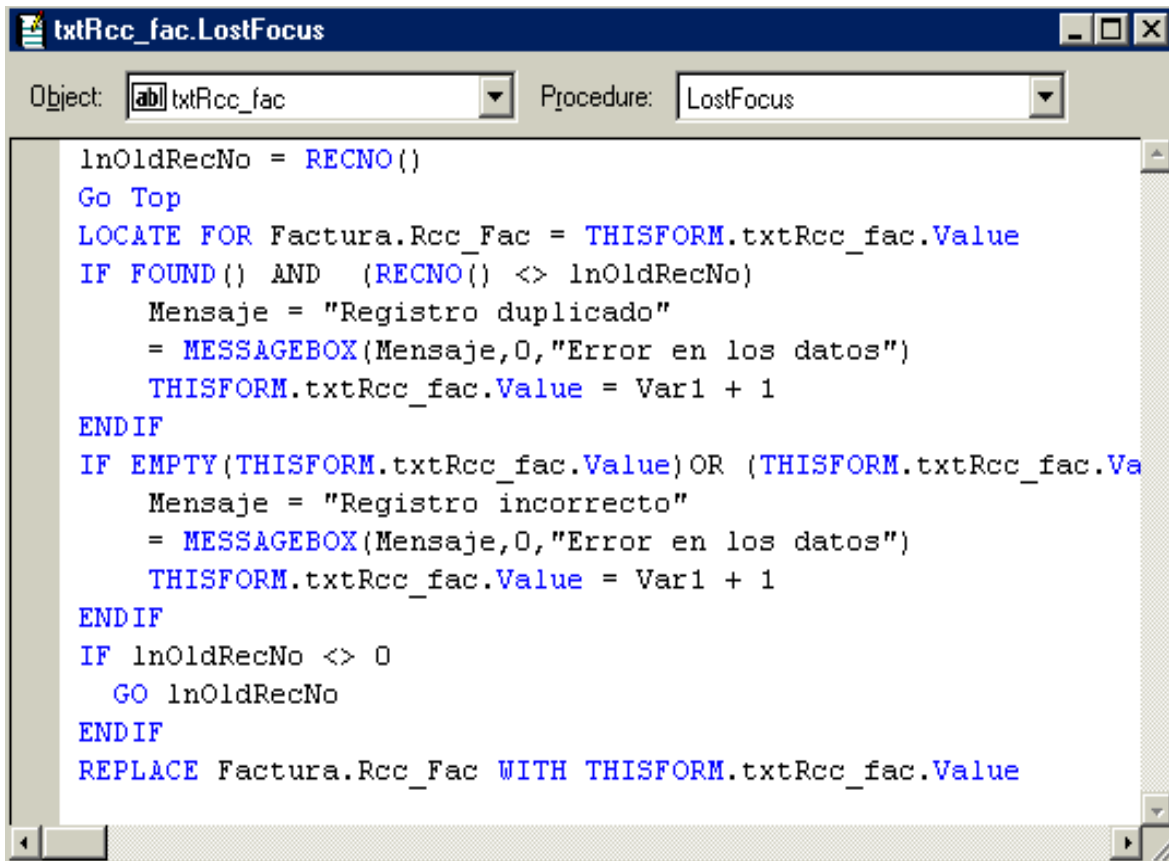


Fig. 6.3.4 Facturación Unitaria de clientes

De este formulario vamos a escoger el objeto de *factura* (txtRcc_fac), el cual emite el número de la factura que se va a modificar o a reimprimir, haciendo una búsqueda de esta factura en la base de datos del sistema para su utilización, mostrándose el siguiente código en el evento LostFocus



```
txtRcc_fac.LostFocus
Object: txtRcc_fac Procedure: LostFocus

lnOldRecNo = RECNO()
Go Top
LOCATE FOR Factura.Rcc_Fac = THISFORM.txtRcc_fac.Value
IF FOUND() AND (RECNO() <> lnOldRecNo)
    Mensaje = "Registro duplicado"
    = MESSAGEBOX(Mensaje,0,"Error en los datos")
    THISFORM.txtRcc_fac.Value = Var1 + 1
ENDIF
IF EMPTY(THISFORM.txtRcc_fac.Value) OR (THISFORM.txtRcc_fac.Va
    Mensaje = "Registro incorrecto"
    = MESSAGEBOX(Mensaje,0,"Error en los datos")
    THISFORM.txtRcc_fac.Value = Var1 + 1
ENDIF
IF lnOldRecNo <> 0
    GO lnOldRecNo
ENDIF
REPLACE Factura.Rcc_Fac WITH THISFORM.txtRcc_fac.Value
```

Fig. 6.3.5 Código en Visual Fox Pro

Al correr la aplicación se despliega el formulario de *facturas unitarias* en el cual se puede modificar o reimprimir una factura seleccionándola de la base de datos, ya que se selecciono la factura se da click en el botón *editar* para poder llevar a cabo los cambios a la misma o simplemente modificarla para su reimpresión normal. Se muestra esto en la siguiente figura.

MODULO DE FACTURACION UNITARIA. FORMA I

Rizo, Armida y Marroquin, S.C.

Factura: 8667 Cliente: INSTALACIONES RIVEL, S.A. DE C.V.

Mes: Febrero Fecha de Emision: 21/02/2002 Estatus: Pendiente Fecha de Pago: / /

Categoría	Concepto	Importe
Gastos	Gastos efectuados conforme al desglose que se acompaña a la presente factura.	\$ 3,799.64

Importe Total (con letra)
 (Son cuatro mil trescientos sesenta y nueve pesos 59/100 M.N.)

Observaciones

Subtotal	\$ 3,799.64
IVA	\$ 569.94
TOTAL	\$ 4,369.58

Fig. 6.3.6 Edición de facturas

7. PRUEBAS Y MANTENIMIENTO DEL SISTEMA

Esta sección considera la interacción del sistema para probar su funcionamiento a través de pruebas de software que tienen como finalidad hacer un examen final del sistema, también se menciona el mantenimiento del sistema, presentado después de la implementación del mismo.

7.1 Pruebas

La prueba del software se considera como una parte crítica de garantía de la calidad del software, representa una revisión final del análisis, diseño y desarrollo de la ingeniería del software. Esta representa entre el 30 y 40 por ciento del esfuerzo del desarrollo de un proyecto.

Objetivos de la Prueba

La prueba no puede asegurar la ausencia de defectos, sólo puede demostrar que existen defectos en el software. Entonces:

- La prueba es un proceso de ejecución de un programa con la intención de descubrir un error.
- Un buen caso de prueba es aquel que tiene una alta probabilidad de mostrar un error no descubierto hasta entonces.
- Una prueba tiene éxito si descubre un error no detectado hasta entonces.

Atributos de una prueba

- Una prueba tiene una alta probabilidad de encontrar un error.
- Una prueba no debe ser redundante.

- Una prueba debería ser la más adecuada.
- Una prueba no debería ser ni demasiado sencilla ni demasiado compleja.

Principios de la Prueba

- A todas las pruebas se les debería de poder hacer un seguimiento hasta los requisitos del cliente.
- Las pruebas deberían planificarse mucho antes de que empiecen.
- El principio de Pareto es aplicable a la prueba del software: “El 80 por ciento de todos los errores descubiertos durante las pruebas surgen al hacer un seguimiento de sólo el 20 por ciento de todos los módulos del programa”.
- Las pruebas deberían empezar por “lo pequeño” y progresar hacia “lo grande”.
- No son posibles las pruebas exhaustivas.
- Para ser más efectivas las pruebas deberían ser conducidas por un equipo independiente.

Por más efectivas nos referimos a pruebas con la más alta probabilidad de encontrar errores.

Facilidad de Prueba

La facilidad de prueba del software es simplemente, lo fácil que se puede probar un programa de computadora.

La siguiente lista de comprobación permite verificar si un software es fácil de probar:

Operatividad: Cuando mejor funcione, más eficientemente se puede probar.

Observabilidad: Lo que ves es lo que pruebas.

Controlabilidad: Cuanto mejor podamos controlar el software, más se puede automatizar y optimizar.

Capacidad de Descomposición: Controlando el ámbito de las pruebas, podemos aislar más rápidamente los problemas y llevar a cabo mejores pruebas de regresión.

- El sistema de software está construido con módulos independientes.
- Los módulos del software se pueden probar independientemente.

Simplicidad: Cuanto menos haya que probar, más rápidamente podremos probarlo.

Estabilidad: Cuanto menos cambios, menos interrupciones a las pruebas.

- Los cambios del software son poco frecuentes.
- Los cambios del software están controlados.
- Los cambios del software no invalidan las pruebas existentes.
- El software se recupera bien de los fallos.

Facilidad de Comprensión: Cuanta más información tengamos, más inteligentes serán las pruebas.

- El diseño se ha entendido perfectamente.
- Las dependencias entre los componentes internos, externos y compartidos se han entendido perfectamente.
- Se han comunicado los cambios del diseño.

- La documentación técnica es instantáneamente accesible.
- La documentación técnica está bien organizada.
- La documentación técnica es específica y detallada.
- La documentación técnica es exacta.

Diseño de casos de Prueba

El diseño de pruebas para el software puede requerir tanto esfuerzo como el propio diseño inicial del producto.

Se debe diseñar pruebas que tengan la mayor probabilidad de encontrar el mayor número de errores con la mínima cantidad de esfuerzo y tiempo posible.

Cualquier producto de ingeniería puede comprobarse de una de estas dos formas:

- Conociendo la función específica para la que fue diseñado el producto, se pueden llevar a cabo pruebas que demuestren que cada función es completamente operativa, y al mismo tiempo buscando errores en cada función. (Prueba de caja negra).
- Conociendo el funcionamiento del producto, se pueden desarrollar pruebas que aseguren que “todas las piezas encajan”, o sea, que la operación interna se ajusta a las especificaciones y que todos los componentes internos se han comprobado de forma adecuada. (Prueba de caja blanca).

Prueba de Caja Negra

Las pruebas de caja negra se centran en los requisitos funcionales del software. La prueba de caja negra permite al Ingeniero de Software obtener conjuntos de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa.

La prueba de caja negra intenta encontrar errores de las siguientes categorías:

- Funciones incorrectas o ausentes.
- Errores de interfaz.
- Errores en estructuras de datos o en accesos a bases de datos externas.
- Errores de rendimiento.
- Errores de inicialización y de terminación.

Prueba de Caja Blanca

La prueba de caja blanca, denominada también prueba de caja de cristal es un método de diseño de casos de prueba que usa la estructura de control del diseño procedimental para los casos de prueba.

La prueba de caja blanca permite al Ingeniero de Software obtener casos de prueba que:

- Garanticen que se ejercita por lo menos una vez todos los caminos independientes de cada módulo.
- Ejerciten todas las decisiones lógicas en sus vértices verdadero y falso.
- Ejecuten todos los lazos en sus límites y con sus límites operacionales.
- Ejerciten las estructuras internas de datos para asegurar su validez.

¿Por qué emplear tiempo y energía probando las minuciosidades lógicas de un Software? Esto se puede responder con lo siguiente:

- Los errores lógicos y las suposiciones incorrectas son inversamente proporcionales a la probabilidad de que se ejecute un camino del programa.
- A menudo creemos que un camino lógico tiene pocas posibilidades de ejecutarse cuando, de hecho, se puede ejecutar de forma normal.
- Los errores de digitación (escritura) son aleatorios.

La aplicación de casos de pruebas apropiadas es esencial para la validación y verificación del sistema construido. Las pruebas normalmente deberían ocupar cerca del 40% del tiempo total de desarrollo de una aplicación. Aún así, no puede asegurarse un 100% de fiabilidad para el sistema. En sistemas donde la fiabilidad y corrección del software es un aspecto crítico las pruebas deben ser más exhaustivas. En estos casos pueden aplicarse también *pruebas de comparación*.

Las herramientas que reducen los tiempos de prueba son muy valiosas. Se pueden distinguir varias categorías de herramientas de prueba: analizadores estáticos, auditores de código, generadores de archivos de prueba, generadores de datos de prueba y controladores de prueba.

Para comprobar la eficacia de este proyecto se empleo como base la prueba de caja negra, aplicando los recursos para probar que tan factibles son los requerimientos funcionales del software, de esta forma determinar errores en procedimientos como:

- Una interfaz que tal vez no es tan amigable con el usuario.
- Una posible falla a la hora de inicializar el sistema
- Funciones ausentes y que necesitan incorporarse al sistema para que este sea más eficiente.

- Funciones incorrectas
- Errores de interfaz.
- Errores en estructuras de datos o en accesos a bases de datos externas.
- Errores de rendimiento.
- Errores de inicialización y de terminación.

7.2 Seguridad

La seguridad de software aplica los principios de la seguridad de información al desarrollo de software. La seguridad de información se refiere comúnmente como la protección de sistemas de información contra el acceso desautorizado o la modificación de información, si esta en una fase de almacenamiento, procesamiento o tránsito. También la protege contra la negación de servicios a usuarios desautorizados, incluyendo las medidas necesarias para detectar, documentar y contrarrestar tales amenazas.

Muchas preguntas con respecto a la seguridad, están relacionadas con el ciclo vital del software. En particular, la seguridad del código y el proceso de software; deben de ser considerados durante la fase del diseño y desarrollo. Además, la seguridad debe de ser preservada durante la operación y el mantenimiento para asegurar la integridad de una parte (pedazo) del software. La falta de seguridad se origina de dos problemas fundamentales: los sistemas que son teóricamente seguros pueden ser inseguros en la práctica, además los sistemas son cada vez más complejos. La complejidad proporciona más oportunidades para los ataques.

Es mucho más fácil probar que un sistema es inseguro que demostrar que uno es seguro, probar la inseguridad, simplemente una toma ventaja de ciertas vulnerabilidades del sistema. Por otra parte, probando un sistema seguro, requiere demostrar que todas las hazañas posibles puedan ser defendidas contra (muy desalentadora), si no imposible, la tarea. Actualmente, no hay ninguna solución singular para asegurar la ingeniería de software. Sin embargo, hay métodos específicos que mejoran la seguridad de los sistemas. En particular, podemos mejorar la confiabilidad de software. También podemos mejorar nuestra comprensión de los requisitos de un pedazo de software.

La confiabilidad de software significa que un programa particular debe de seguir funcionando con la presencia de errores. Los errores pueden ser relacionados al diseño, a la implementación, a la programación o el uso de errores. Así como los sistemas llegan a ser cada vez más complejos, aumenta la probabilidad de errores. Como mencionamos, es increíblemente difícil demostrar que un sistema sea seguro. Software seguro debe de funcionar bajo un ataque. Aunque casi todo el software tenga errores, la mayoría de los

errores nunca serán revelados debajo de circunstancias normales. Un atacante busca esta debilidad para atacar un sistema.

La arquitectura de la base de datos del estándar Fox Base no incluye restricciones de acceso o encriptación de información a nivel registro, para esto es necesario activar “triggers” (evento disparador que se genera a partir de código insertado en la estructura de la base de datos) lo cual ameritaría aumentar el tiempo de desarrollo del sistema. Por lo consiguiente se incrementa el costo y quedaría fuera de mercado.

En lo referente a la seguridad de acceso al sistema se establecieron perfiles de usuarios destinados exclusivamente al uso del mismo, los usuarios pertenecen a una categoría de acceso la cual delimita los módulos que pueden acceder y/o obtener reportes e información, determinando la jerarquía de cada usuario para el acceso completo o parcial a los módulos principales del sistema, diferenciando de esta forma la accesibilidad a partes críticas dentro del sistema que pueden ser cruciales para el manejo óptimo del mismo y que es de vital importancia que solo se tenga acceso a través de cuentas específicas previamente determinadas, definiéndose esto como sigue:

- Administrador del Sistema. Acceso total al sistema, persona encargada de crear o supervisar las cuentas existentes del sistema.
- Socios. Acceso total al sistema, con asesoría del Administrador del Sistema.
- Personal Administrativo. Personal encargado de administrar y capturar la información del sistema, previa autorización del Socio puede tener acceso a todos los módulos del sistema.

7.2 Mantenimiento del sistema.

La fase de mantenimiento se centra en el cambio que va asociado a la corrección de errores, a las adaptaciones requeridas a medida que evoluciona el entorno del software y a cambios debidos a las mejoras por los requisitos cambiantes del cliente. El tiempo de uso de un sistema trae consigo la necesidad de mantenerlo continuamente.

En estudios realizados se ha concluido lo siguiente:

- Del 60% al 90% del costo total del software durante la vida útil de un sistema se gasta en mantenimiento.
- En algunos casos, el costo del mantenimiento es más de 50 veces el costo inicial de desarrollo del sistema.
- Usualmente 2/3 partes del personal técnico dedican su tiempo al mantenimiento del software.

Tan solo el 20 por ciento de nuestros esfuerzos de mantenimiento se invertirán corrigiendo errores. El 80 por ciento restante se dedica a adaptar los sistemas existentes a los cambios de su entorno externo, a efectuar las mejoras solicitadas por los usuarios y a rehacer la ingeniería de las aplicaciones para su posterior utilización.

Tipos de Mantenimiento

Mantenimiento implica más que la "corrección de errores." Pueden describirse 4 diferentes tipos de mantenimiento: correctivo, adaptativo, perfectivo y preventivo.

Mantenimiento Correctivo

Involucra el diagnóstico y corrección de uno o más errores. Ocurren errores en el sistema que el usuario reporta al desarrollador. Una vez instalados los sistemas, la necesidad de corregir errores de emergencia es relativamente baja (menos del 20% de las tareas son de corrección).

Mantenimiento Adaptativo

El mantenimiento adaptativo modifica el sistema para adecuarlo a los cambios del medio ambiente, (entorno original).

Este mantenimiento se debe a los cambios rápidos que ocurren en cualquier aspecto computacional. Por ejemplo, nuevas versiones del software de desarrollo, de sistemas operativos, de equipo periférico, etcétera.

El mantenimiento de los sistemas también implica adaptaciones de versiones anteriores, adecuar cambios de reportes, archivos y procesos (aproximadamente un 20%).

Mantenimiento Perfectivo (Mejoras)

Conforme el sistema es usado, nuevas recomendaciones para satisfacer nuevos requerimientos, modificaciones a las funciones existentes y mejoras en general son recibidas por parte de los usuarios.

La mayor parte del tiempo de mantenimiento se convierte en perfectivo conforme pasa la vida del sistema.

Este tipo de mantenimiento trata de cumplir las nuevas peticiones de los usuarios, mejorar la documentación o decodificar para mejorar la eficiencia (por ejemplo: tiempo de respuesta). Aproximadamente el 40% de todo el mantenimiento tiene este fin.

Mantenimiento Preventivo (Reingeniería del Software)

El software se deteriora debido al cambio, por ello deben efectuarse cambios en el software a fin de que se pueda corregir, adaptar y mejorar más fácilmente (facilidad de mantenimiento), a este proceso se le llama Reingeniería del Software. Este tipo de mantenimiento usualmente es sugerido por el mismo departamento de desarrollo. Ocupa aproximadamente el 20% del tiempo de desarrollo.

Es indudable que el software una vez entregado al cliente sufrirá cambios (posible excepción es el software empotrado). Los cambios ocurrirán debido a que se hayan encontrado errores, a que el software deba adaptarse a posibles ajustes, etcétera.

El desarrollo de productos de software bajo éste ciclo de vida tiene los siguientes problemas:

- Los proyectos reales raramente siguen el flujo secuencial que propone el modelo.
- Normalmente, es difícil para el cliente establecer explícitamente al principio todos los requisitos. Este ciclo lo requiere y tiene dificultades en entender posibles problemas que pudieran existir.
- El cliente debe tener paciencia. Hasta llegar a las etapas finales del desarrollo del proyecto.

A pesar de sus inconvenientes, el ciclo de vida en cascada sigue siendo el modelo clásico más ampliamente usado por los ingenieros del software.

El mantenimiento se puede llevar hasta un 70% de esfuerzo de desarrollo, por lo que es necesario analizar adecuadamente las acciones a realizar con el propósito de minimizar los probables errores o cambios a corregir, pudiéndose clasificar este en dos grandes tipos Correctivo y Preventivo.

El mantenimiento es una reindexación de registros en la base de datos, depuración de registros eliminados, respaldo de información, ajustes en reportes, administración de usuarios, etcétera.

CONCLUSIONES

La elaboración de cualquier trabajo final, en especial un trabajo de tesis, es una experiencia inolvidable, esto se proyecta a través del contenido de todas estas paginas, que representan el diseño y desarrollo de un sistema computacional para la administración y control de cuentas por cobrar de los despachos jurídicos, con una interfaz sencilla y robusta que permita optimizar las operaciones de este tipo de organizaciones.

La realización de esta tesis tomó un tiempo aproximado de 7 meses, al inicio se recopilaron los datos necesarios para llevar a cabo este proyecto, pudiendo mencionar que la mayor parte del proceso de elaboración del sistema se ocupó en la programación y las pruebas del sistema, que es donde se detectaron los problemas o errores funcionales en tiempo de ejecución y que se pudieron resolver cuando se realizó el mantenimiento del mismo.

Para el proceso de esta tesis nos vinculamos directamente hacia el sector de micro y pequeñas empresas del área de servicios que involucran principalmente a los despachos jurídicos, obteniendo y recabando información necesaria para desarrollar un sistema acorde con las necesidades de facturación y cobranza que presentan este tipo de organizaciones.

El proyecto sirvió para recordar los conocimientos adquiridos a lo largo de la carrera, pero principalmente para desarrollar habilidades que la licenciatura de Ingeniería en Computación nos marca como pauta para llegar a ser excelentes profesionistas, comprometidos con nuestro entorno y con la sociedad. Las materias que se utilizaron como base para lograr esta tesis, fueron: computadoras y programación, estructuras de datos, bases de datos y las materias optativas de evaluación de proyectos y calidad, principalmente.

En general los despachos jurídicos, tienen un proceso de emitir facturas por medio de hojas de cálculo, principalmente la de Excel de la empresa Microsoft, lo que origina mucho consumo de tiempo de los recursos humanos y materiales de la organización, lo

que motiva que los costos de operación sean altos. El personal que se encarga de realizar este trabajo comúnmente no tiene conocimientos acerca del uso de herramientas de software para llevar el control de cuentas por cobrar de un despacho de abogados (bases de datos, principalmente), por lo que primeramente se tuvo que franquear esta barrera de usuarios que tenían una herramienta que “funcionaba” para la operación que requerían, pero en realidad era impráctica y corta de resultados, este fue uno de los retos más difíciles de afrontar principalmente para convencer y acceder el proyecto a los usuarios finales.

Con estas características mencionadas se propuso un sistema que hiciera más eficaz este proceso, teniendo como base los siguientes objetivos en mente:

- Proporcionar un sistema computacional que permita organizar el proceso de cobros de los clientes de los despachos jurídicos.
- Obtener una herramienta computacional práctica y confiable para la administración de cobros de los despachos jurídicos
- Crear un sistema de cómputo con soluciones sencillas y robustas, de manera que permita al usuario final obtener beneficios a corto y largo plazo, de esta forma consolidar toda la información que se genere de esto.

Para la elaboración de esta tesis se tomó como base la metodología del ciclo de vida clásico en cascada, lo que se buscó con esto, es proporcionar al usuario una tecnología mas amigable y segura que le permita hacer más eficiente los procesos de facturación y cobro de los despachos jurídicos.

El desarrollo del sistema ha optimizado enormemente la función de facturación de los despachos jurídicos donde fue implantado, mejorando la eficiencia y la eficacia de los procesos de cobros y por consiguiente mejorando la administración de los ingresos de los mismos.

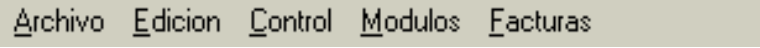
Es importante recalcar la relevancia de las tecnologías de información en el desarrollo de cualquier empresa u organización privada o pública, lo que representa un sinnfín de

oportunidades para los profesionales de estas áreas. Como Ingenieros en Computación debemos desarrollar destrezas no sólo de carácter técnico, sino también habilidades de negociación de esta manera uno pueda colocarse más rápida y eficientemente en el mercado laboral que cada día es más competido. En este sentido el haber estudiando en una institución como la UNAM, donde se apoya el crecimiento integral de todos sus estudiantes a través de diferentes y variadas áreas de conocimientos (profesionales, culturales, deportivas, recreativas, etcétera) nos permite fomentar y estimular tanto experiencias técnicas, como de desarrollo personal o cultural.

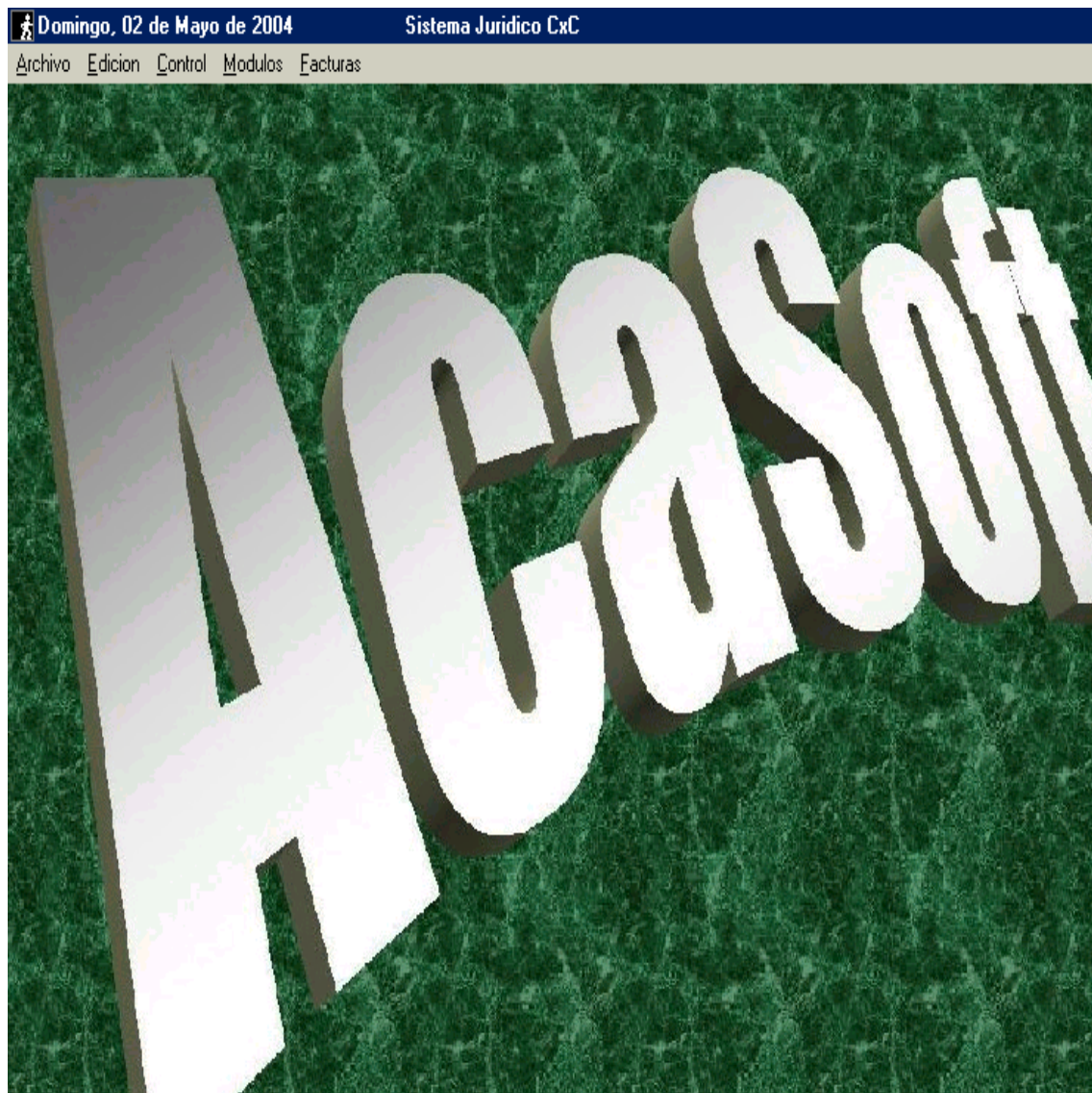
Finalmente podemos decir, que el tiempo y recursos empleados para la realización de este proyecto han sido aprovechados de manera satisfactoria de acuerdo con los objetivos planteados al inicio de esta tesis y de esta forma se ha podido optimizar los procesos de administración y control de las cuentas por cobrar de los despachos jurídicos analizados para la realización de esta tesis.

El manual sirve como apoyo para que un usuario con conocimientos básicos de computación pueda utilizar apropiadamente las herramientas que proporciona el Sistema Jurídico de Cuentas por Cobrar (SJCC), a continuación se muestra un instructivo de la estructura general del proyecto.

En primera instancia se presenta la pantalla principal de desarrollo de SJCC, en la cual se observa como menú principal lo siguiente:

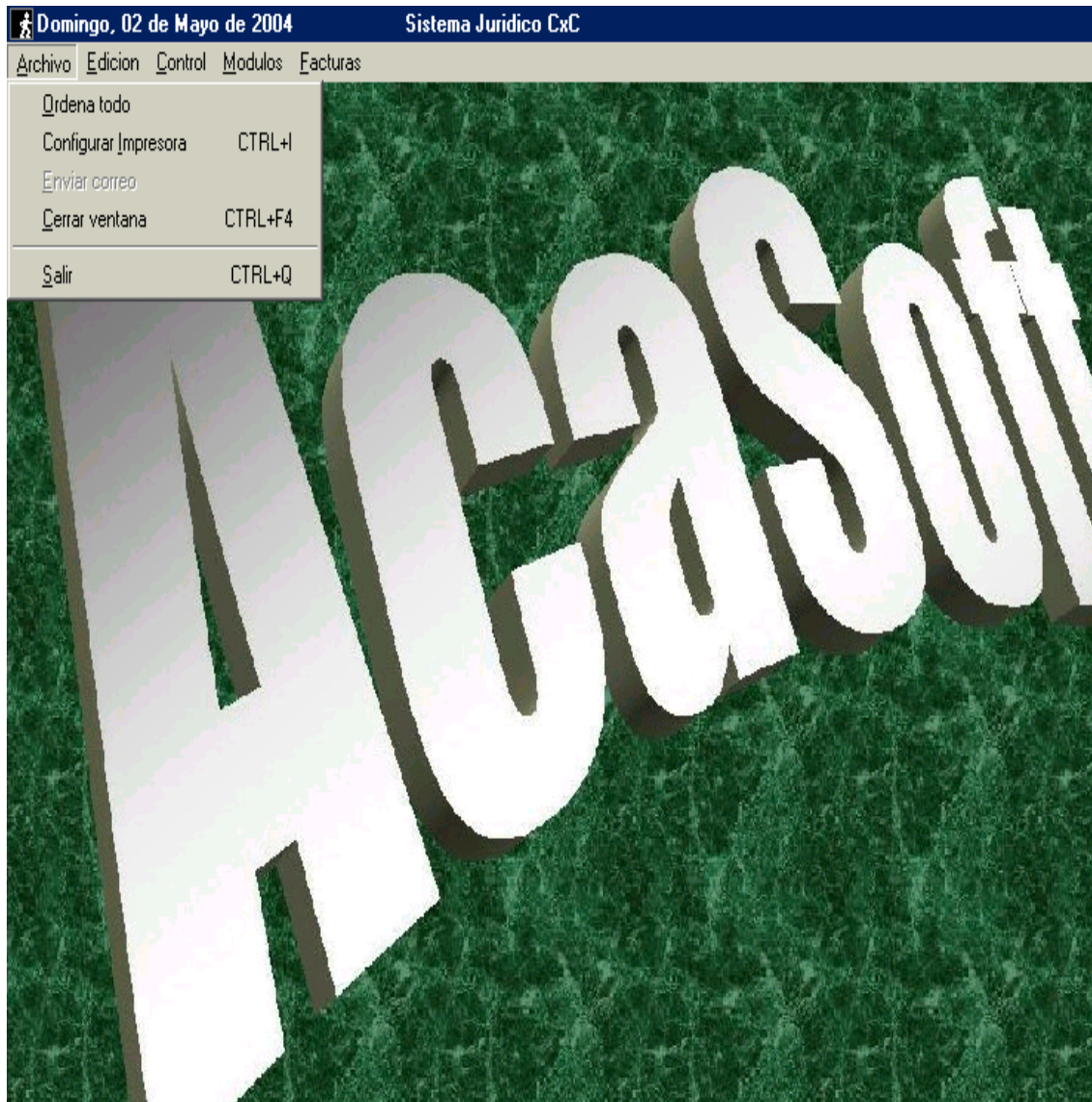


Primero se presenta la pantalla completa del sistema, de acuerdo al modelo escogido para su desarrollo, desglosando cada uno de los módulos que conforman a los diferentes menús del Sistema Jurídico de Cuentas por Cobrar.



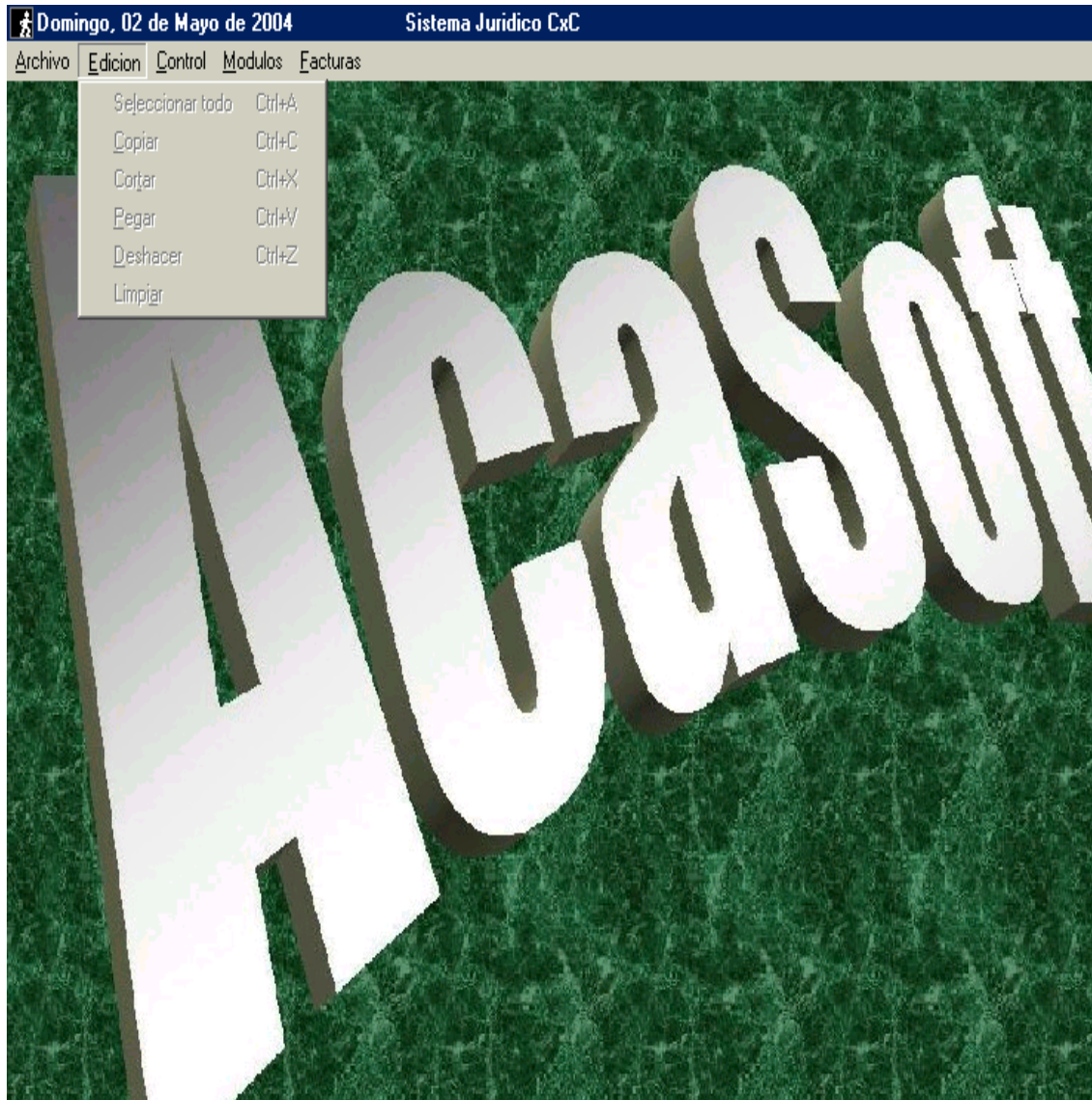
Se presentan a continuación los diversos módulos del sistema desglosándose los diferentes comandos que conforman a cada uno de los menús.

MENÚ ARCHIVO. Se conforma de los siguientes comandos.



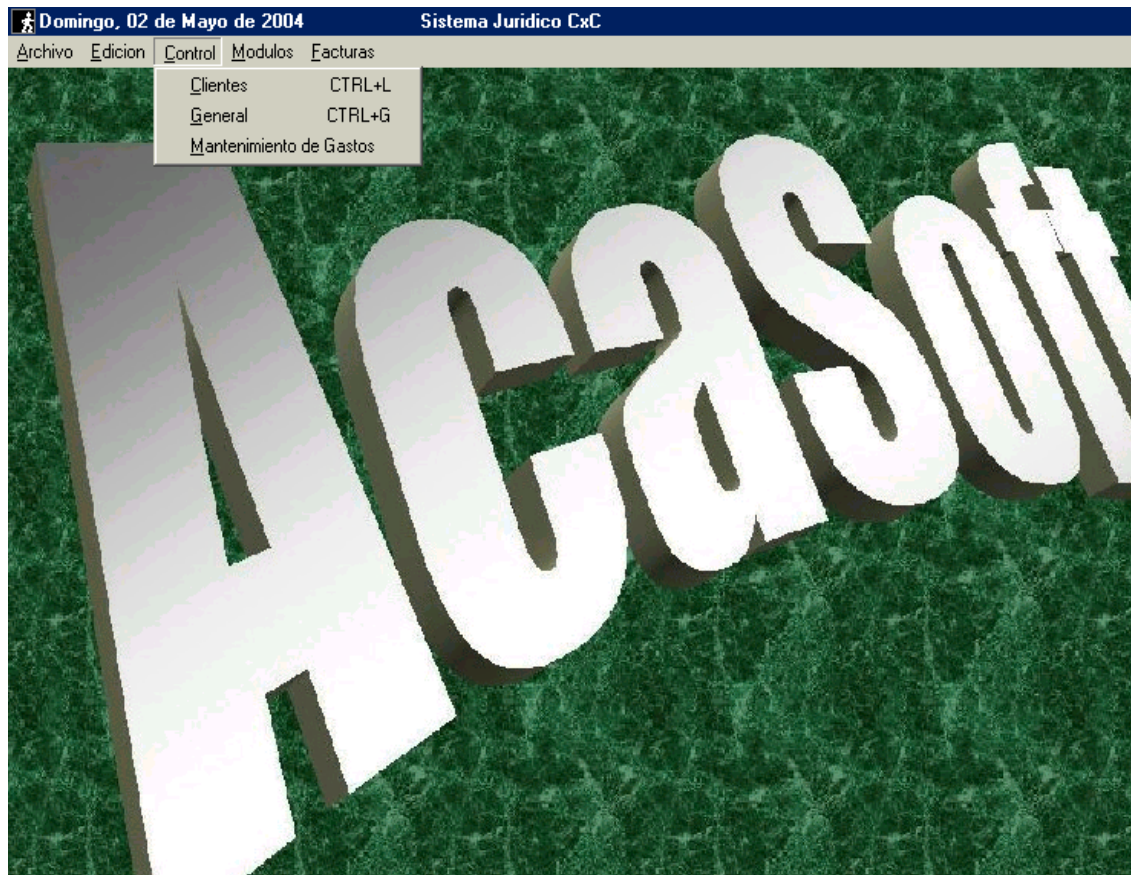
Este menú se utiliza de forma igual al que aparece en Windows para el manejo de archivos dentro de sus sistemas, aunque en este menú la principal función se determina para *configurar impresora*, es decir, indicar que impresora se ocupará, que tamaño de papel, posición de la hoja, etcétera. Esta función también se puede activar con las teclas (CTRL+I).

MENÚ EDICIÓN. Es un menú similar al que ocupan los archivos de Windows para edición y se compone de los siguientes comandos:



Al igual que el menú anterior, éste es un menú similar al ocupado por Windows para el manejo de archivos, teniendo los siguientes comandos: *seleccionar todo*, que sirve para seleccionar todo el texto o la que aparezca en la pantalla al momento de la selección, el comando *copiar* para copiar algún archivo o selección hecha con anterioridad, el comando *cortar* para cortar algún archivo o selección hecha con anterioridad, el comando *pegar* para pegar la selección o archivo copiado o cortado, el comando *deshacer* que sirve para corregir una acción anterior y el comando *limpiar* que se utiliza para limpiar los campos de alguna ventana previamente abierta.

MENÚ CONTROL. Es un menú segmentado de la siguiente manera:



Formulario Clientes (CTRL+L).

CATALOGO DE CLIENTES X

Rizo, Armida y Marroquin, S.C.

N°de Cliente	Razon Social	Lista de clientes			
1	ALIJUMEX, S.A. DE C.V.	▼			
R.F.C.	Calle y No.	Colonia			
AJU 910107PX8	Boulevard Morelos 72-A	Rústica Xalostoc			
Ciudad	Estado	Codigo Postal	Telefono 1	Telefono 2	Fax
Ecatepec	México	55340			
▼	▼	Correo Electronico		Moneda	
				M.N. ▼	
Observaciones					
Iguala mensual	Proveedor	Tarifas			
\$22,754.00		Socio	Asociado	Pasante	
		\$250.00	\$200.00	\$150.00	

✖ ✎ 💾 📄 + - ↶ 🔍 ⏪ ⏩ ⏴ ⏵ 🔍 🔌
 Borrar Editar Grabar Nuevo Línea Línea Deshacer Buscar Primero Anterior Siguiente Ultimo Vista Cerrar

En este módulo se pueden dar de alta todos los datos de facturación de los clientes, modificar sus datos, dar de baja el cliente, etcétera. Así mismo se puede registrar que tipo de facturación se va a efectuar: por iguala, por honorarios o por trabajo realizado.

Formulario General (CTRL+G). En este módulo se tienen valores (etiquetas) que se utilizan en los diferentes módulos del sistema. Las secciones son las siguientes:

Generales. Meses del año, instituciones bancarias, estatus de facturas, tipos de pagos, campos adicionales con un valor asociado en la columna varios.



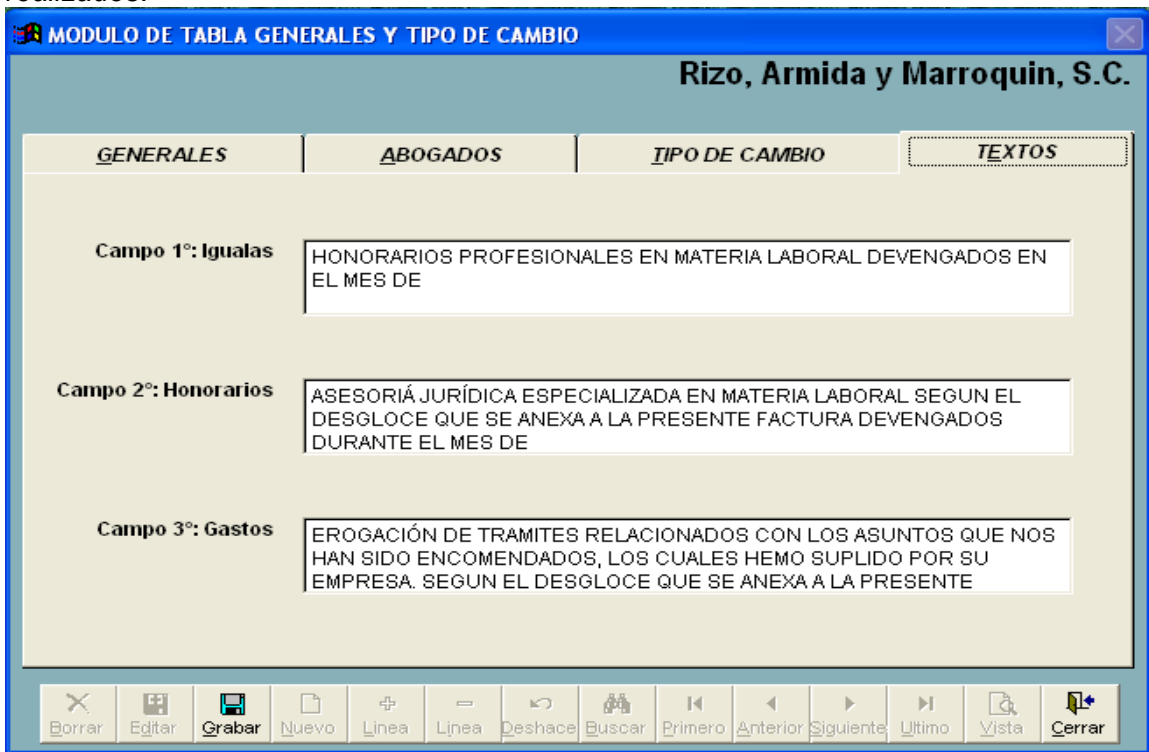
Abogados. Nombre de los usuarios a los que se les registran operaciones de gastos y honorarios en el sistema, los cuales son clasificados según el valor asignado en la columna "Autorizados", esta clasificación asocia al usuario con la tarifa dada de alta en el Formulario de Clientes.



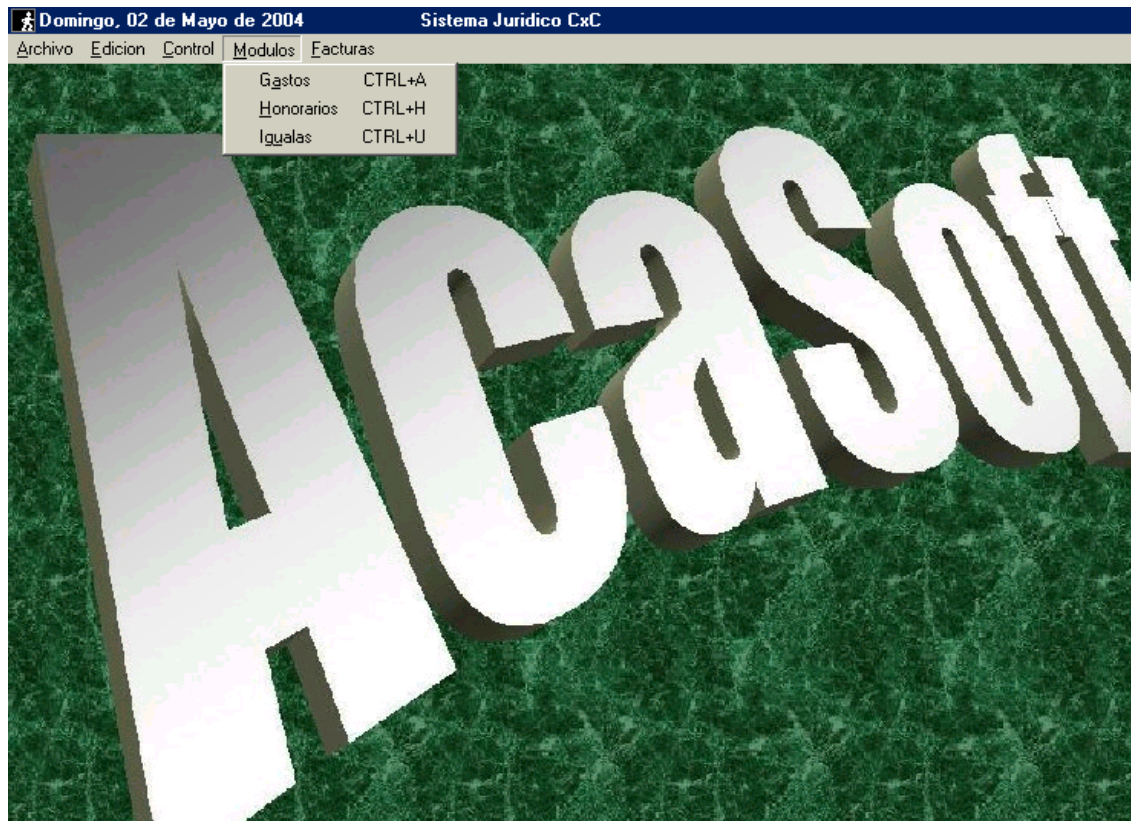
Tipo de cambio. Dos registros; “Fecha” es la fecha en la cual se esta realizando la facturación al cliente, y “Tipo de cambio”, si se maneja pago en otra moneda diferente al peso, se toma como referencia esta columna para especificar en que tipo de moneda se esta efectuando el pago.



Textos. Se manejan tres secciones diferentes de texto que se presentarán en el reporte o factura mostrada al cliente de acuerdo al concepto que se le este cobrando, conforme al tipo de facturación que se realice, *por iguales, por honorarios o por gastos realizados.*

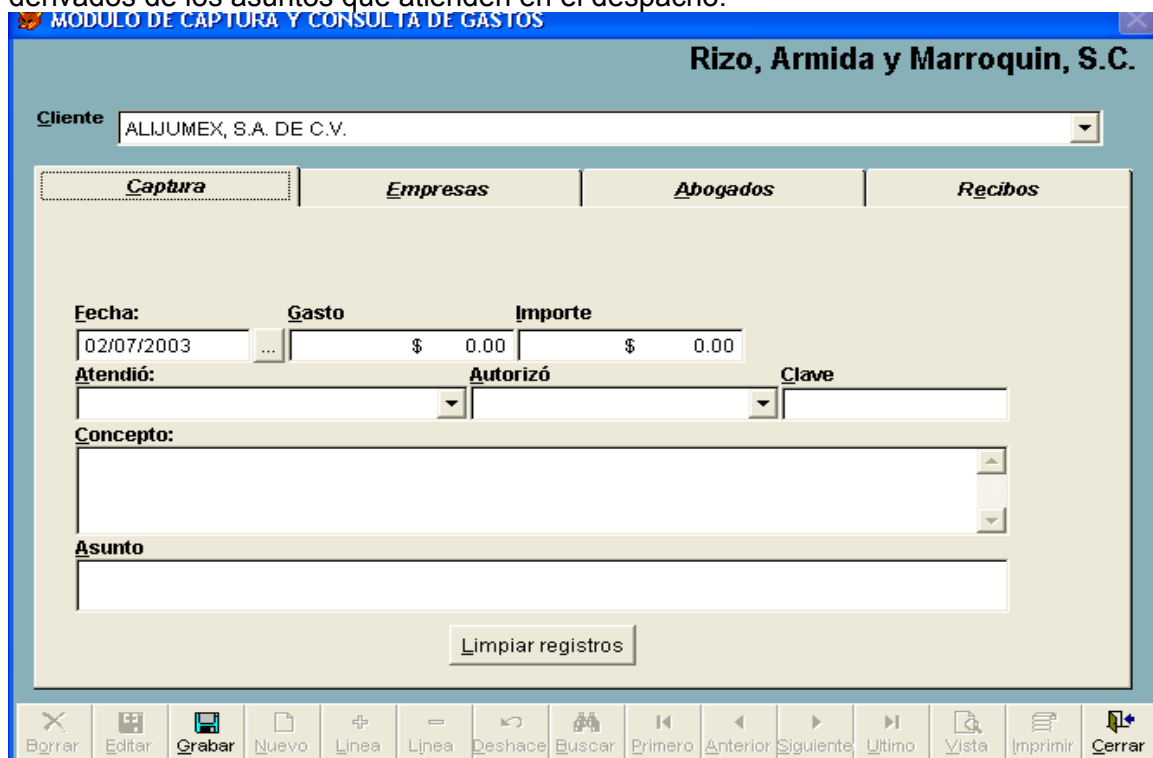


MENÚ MÓDULOS. Este menú se segmenta de la siguiente manera:



Formulario Gastos (CTRL+A). Este módulo se encarga de capturar y consultar los gastos generados por atender los asuntos legales de los clientes es decir, papelería, gastos de traslado, gastos generales, etcétera. Desglosándose de la siguiente manera:

Captura. En este módulo se registran los gastos erogados por el personal legal derivados de los asuntos que atienden en el despacho.



Empresas. Muestra en detalle la consulta de los gastos relacionados a una empresa, así como los gastos pagados o adeudados.

The screenshot shows the 'MODULO DE CAPTURA Y CONSULTA DE GASTOS' window for 'Rizo, Armida y Marroquin, S.C.'. The 'Cliente' field is set to 'ALIJUMEX, S.A. DE C.V.'. The 'Empresas' tab is selected, showing a table with columns: Fecha, Factura, Atendió, Asunto, Concepto, and Importe. Below the table, there are filters for 'Mes' (Julio) and 'Año' (2003), and buttons for 'Pendiente' and 'Facturado'. Summary fields show 'SubTotal: \$ 0.00', 'IVA: \$ 0.00', and 'Total: \$ 0.00'. A toolbar at the bottom includes icons for 'Borrar', 'Editar', 'Grabar', 'Nuevo', 'Linea', 'Linea', 'Deshace', 'Buscar', 'Primero', 'Anterior', 'Siguiete', 'Ultimo', 'Vista', 'Imprimir', and 'Cerrar'.

Abogados. En esta sección se realiza el pago al personal legal del despacho de los gastos en que haya incurrido y registrado a nombre del cliente, así como la consulta de los pagos realizados con anterioridad.

The screenshot shows the 'MODULO DE CAPTURA Y CONSULTA DE GASTOS' window for 'Rizo, Armida y Marroquin, S.C.'. The 'Cliente' field is set to 'ALIJUMEX, S.A. DE C.V.'. The 'Abogados' tab is selected, showing a table with columns: Fecha, Abogado, Autorización, Asunto, and Importe. Below the table, there are filters for 'Intervalo' (14/07/2003 al 21/07/2003) and 'Abogado' (Despacho). Buttons for 'Adeudo', 'Pagado', and 'Efectuar pago' are present. Summary fields show 'SubTotal: \$ 0.00'. A toolbar at the bottom includes icons for 'Borrar', 'Editar', 'Grabar', 'Nuevo', 'Linea', 'Linea', 'Deshace', 'Buscar', 'Primero', 'Anterior', 'Siguiete', 'Ultimo', 'Vista', 'Imprimir', and 'Cerrar'.

Recibos. Esta sección es opcional dependiendo de la forma de asociar estos gastos para su pago, ya sea mediante un recibo no fiscal o asociados a una factura como gastos de representación y muestra los recibos generados al cliente seleccionado, así como el detalle de los mismos.

Formulario Honorarios (CTRL+H). En este módulo se registran los eventos a manera de bitácora que genera cada usuario del sistema cuando este atiende alguna actividad relacionada con un cliente y que produce un importe a cobrar al cliente en cuestión, consolidando estos importes a una factura en el proceso de impresión masiva.

Se divide en dos puntos:

Captura. Captura de los honorarios de cada abogado.

Resumen. Reporte de los honorarios a pagar por cliente.

Fecha	Factura	Atendió	Concepto	Solicitado por	Importe

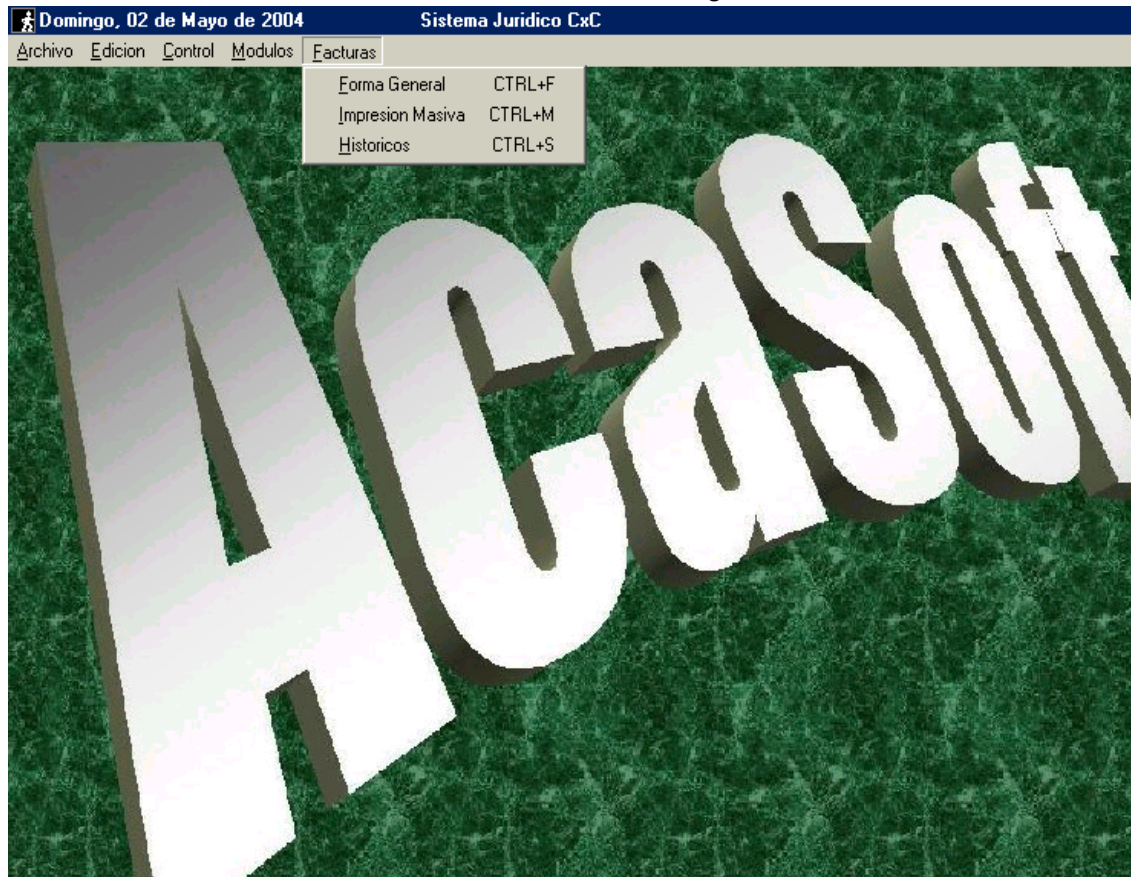
Formulario *Igualas* (CTRL+U). Este módulo muestra el número de clientes existentes que tienen una iguala con la compañía, mostrando el importe total facturado por estas igualas por clientes y ordenados en orden alfabético, en este modulo se puede editar estos campos a conveniencia del usuario, simplemente es un reporte de la totalidad de igualas consideradas por la empresa.

MODULO DE CAPTURA DE IGUALAS Rizo, Armida y Marroquin, S.C.

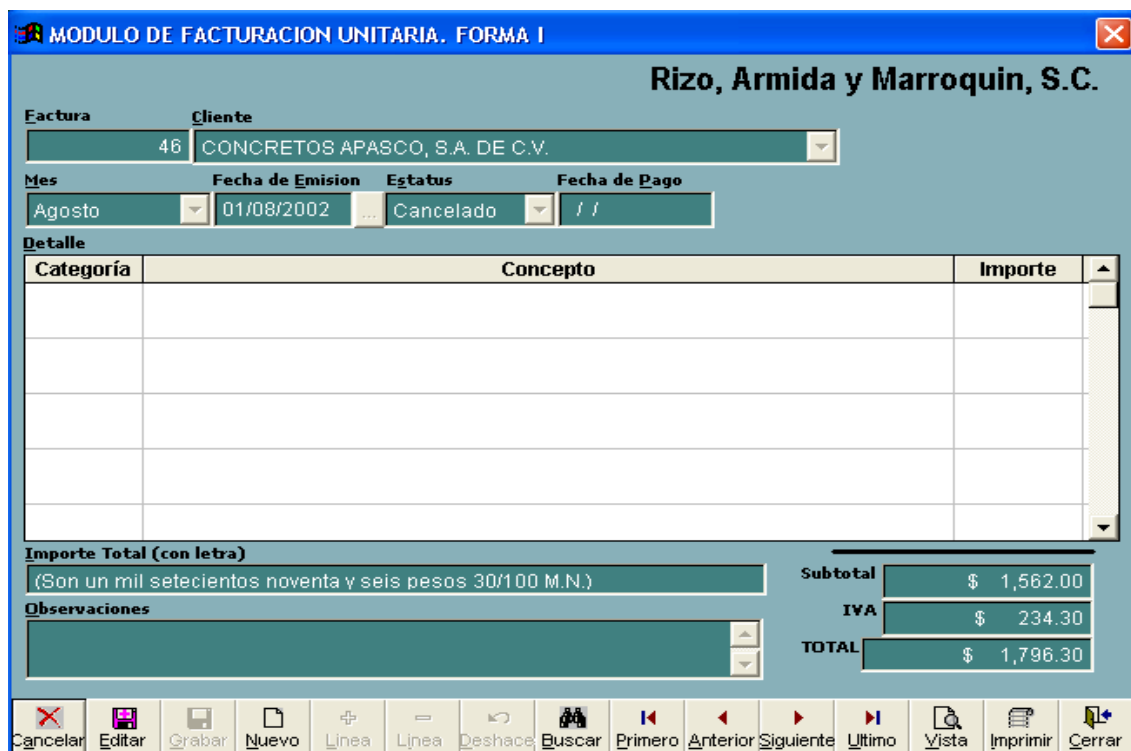
Numero	Razon Social	IMPORTE
1	ALIJUMEX, S.A. DE C.V.	\$22,754.00
2	ASTRAZENECA, S.A. DE C.V.	\$18,000.00
3	BANCO OBRERO, S.A. (EN LIQUIDACION)	\$6,000.00
4	BARCEL, S.A. DE C.V.	\$3,161.25
5	BIENVENIDO AL MUNDO, S.A. DE C.V.	\$2,000.00
6	BRIDGESTONE/FIRESTONE DE MEXICO, S.A. DE C.V.	\$13,000.00
9	CIA. INDUSTRIAL AZUCARERA SAN PEDRO, S.A. DE C.V.	\$6,000.00
12	COMERCIALIZADORA MORESA, S.A. DE C.V.	\$5,940.00
13	COMERCIAL MEXICANA DE PINTURAS, S.A. DE C.V.	\$15,288.00
14	COMEXTRA, S.A. DE C.V.	\$7,000.00
15	CORPORTIVO DINE, S.A. DE C.V.	\$7,000.00
16	CORPORATIVO UNIKO, S.A. DE C.V.	\$1,637.42

Borrar Editar Grabar Nuevo Línea Línea Deshace Buscar Primero Anterior Siguiente Ultimo Lista Cerrar

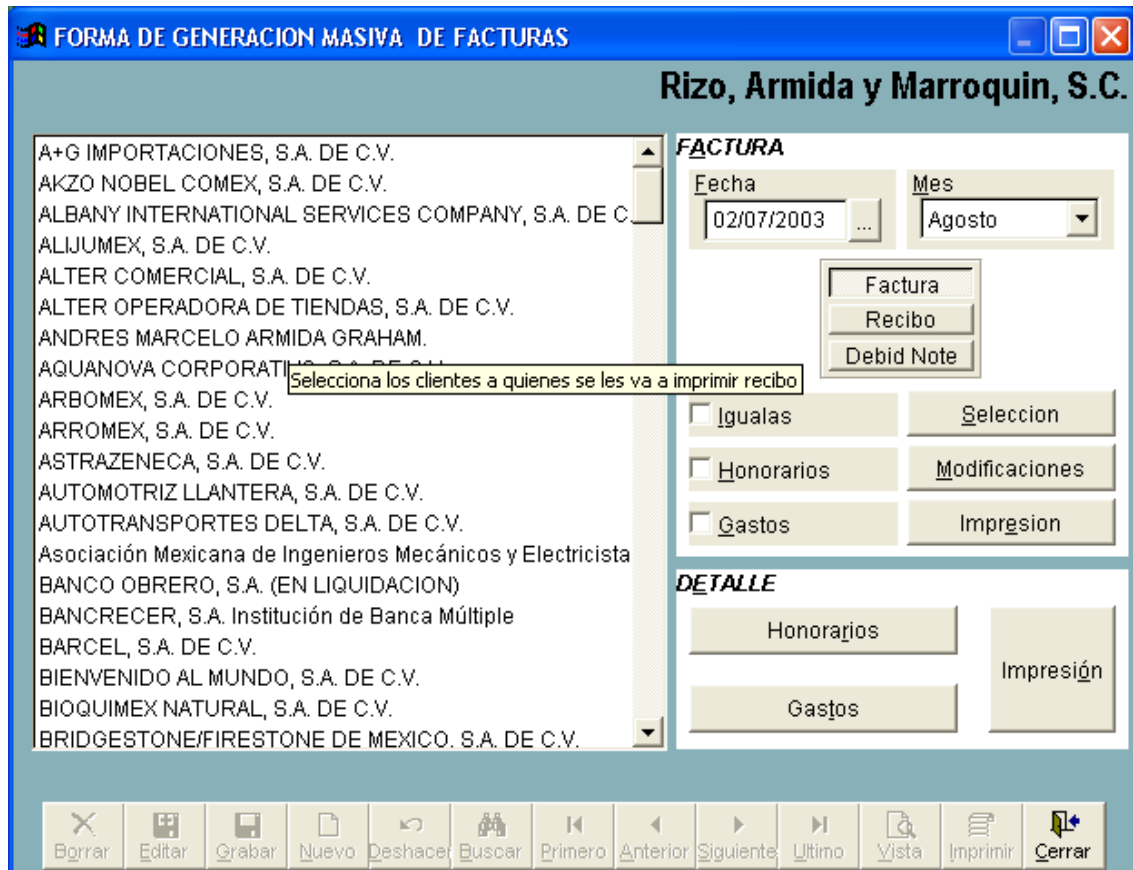
MENÚ FACTURAS. Este menú se conforma de los siguientes comandos:



Formulario **Forma General (CTRL+F)**. Es el módulo de reportes de facturas, reimpresión y también se puede hacer una única factura solicitada por el cliente.



Formulario de Impresión Masiva de Documentos de Cobro (CTRL+M). En este módulo se lleva a cabo la impresión masiva de facturas, recibos, debit notes, o cualquier documento de cobro que necesite tener seguimiento, así como folios consecutivos dependiendo del tipo esquema de pago pactado con el cliente, ya sea honorarios por hora, por obra determinada o por iguala mensual, pudiéndose combinar estos esquemas para generar un documento de cobro que consolide una o varias actividades realizadas en ciertos periodos de tiempo, así mismo en este formulario se pueden generar automáticamente textos de conceptos, importes y detalle de actividades a cobrar.



Formulario de Facturación Histórica (CTRL+S). Aquí se muestran diferentes campos donde se indica a detalle la situación que guardan los clientes con el despacho bajo las siguientes secciones:

Consolidado. Muestra la información más importante de lo facturado, lo adeudado y lo pagado por el cliente.

Consolidado | Facturas | Detalle | Pagos | Reportes

Operaciones en el año

Saldos x Estatus	
Estatus	Acumulado
Pendiente	\$108,073.93

Saldo
\$ 196,262.06

Pagos
\$ 0.00

Operaciones acumuladas

Saldos x Estatus	
Estatus	Acumulado
Cancelado	\$11,500.00
Juridico	\$132,974.16
Pendiente	\$395,682.58

Saldo
\$ -686,379.36

Pagos
\$ 0.00

Saldo a favor \$ 0.00

Facturas. Muestra todas las facturas emitidas al cliente ordenadas cronológicamente.

Consolidado | **Facturas** | Detalle | Pagos | Reportes

Factura	Importe	Mes	Fecha Pago	Fecha Emision	Estatus	Observaciones
9919	\$ 27,601.15	Enero	/ /	02/01/2003	Pendiente	memo

10027	\$ 27,024.77	Febrero	/ /	03/02/2003	Pendiente	memo
10098	\$ 2,865.80	Febrero	/ /	03/02/2003	Pendiente	memo
10023	\$ 17,250.00	Enero	/ /	31/01/2003	Pendiente	memo
10099	\$ 2,635.80	Febrero	/ /	03/02/2003	Pendiente	memo
10153	\$ 30,696.41	Marzo	/ /	03/03/2003	Pendiente	memo

Detalle. Es el desglose de los conceptos de las facturas, las cuales se despliegan al seleccionar la factura deseada en la sección de facturas.

Pagos. En esta forma se registran los pagos de los clientes, asociándose a facturas y/o recibos adeudados.

FACTURACIÓN HISTÓRICA Rizo, Armida y Marroquin, S.C.

Cliente: ALIJUMEX, S.A. DE C.V. Año: 2003

Consolidado Facturas **Detalle** Pagos Reportes

Categoría	Concepto	Importe
9919	Iguala correspondiente al mes de enero de 2003.	22754.0000
9919	Gastos, según relación anexa.	1247.0000

Borrar Editar Grabar Nuevo Linea Deshacer Buscar Primero Anterior Siguiente Ultimo Vista Imprimir Cerrar

Pagos. Aquí se tiene el reporte del estatus de pagos de los clientes del despacho.

FACTURACIÓN HISTÓRICA Rizo, Armida y Marroquin, S.C.

Cliente: ALIJUMEX, S.A. DE C.V. Año: 2003

Consolidado Facturas Detalle **Pagos** Reportes

Importe	Fecha Pago	Tipo de pago	Cheque/Folio	Banco	Observaciones
\$ 0.00	18/07/2003				

Alta de pago

Factura A cuenta 8683

Borrar Editar Grabar Nuevo Linea Deshacer Buscar Primero Anterior Siguiente Ultimo Vista Imprimir Cerrar

Reportes. Esta sección se usa para dar de alta reportes de información solicitada en específico por el cliente del despacho.

The screenshot shows a software window titled "FACTURACIÓN HISTÓRICA" with a close button in the top right corner. The main title of the window is "Rizo, Armida y Marroquin, S.C.". Below the title, there are two input fields: "Cliente" with the value "ALIJUMEX, S.A. DE C.V." and "Año" with the value "2003".

Below these fields is a horizontal menu with five tabs: "Consolidado", "Facturas", "Detalle", "Pagos", and "Reportes". The "Reportes" tab is currently selected and highlighted with a dotted border.

Under the "Reportes" tab, there is a section labeled "Mes" with a dropdown menu. Below this are two buttons: "Pagos" and "Facturas".

At the bottom of the window is a toolbar with the following icons and labels from left to right: "Borrar" (trash icon), "Editar" (edit icon), "Grabar" (save icon), "Nuevo" (new icon), "Linea" (minus icon), "Deshacer" (undo icon), "Buscar" (magnifying glass icon), "Primero" (double left arrow icon), "Anterior" (single left arrow icon), "Siguiente" (single right arrow icon), "Ultimo" (double right arrow icon), "Vista" (magnifying glass icon), "Imprimir" (printer icon), and "Cerrar" (close icon).

BIBLIOGRAFÍA

Menachem Bazian
Visual FoxPro 6, Edición Especial
Pearson Education, México, 2000

Pressman Roger
Ingeniería del software
McGraw Hill, 1998
Cuarta Edición

Abeled-Perrot
Diccionario Jurídico,
Buenos Aires, 1993
Segunda Edición Ampliada.

Mario Piattini Adoración de Miguel
Fundamentos y modelos de base de datos
Editorial Alfaomega.

Henry Korth
Abraham Silberschatz
Análisis y diseño de sistemas
McGraw Hill
Segunda Edición.

Linda Gail
Enciclopedia de Términos de Computación
Prentice Hall.

James A. Senn
Análisis y diseño de sistemas
McGraw Hill, México
Segunda Edición.

- [1] <http://www.monografias.com/trabajos/anaydisis/anaydisis.shtml>
- [2] <http://www.mcc.unam.mx/~cursos/objetos/Cap17/Cap17.html>
- [3] <http://www.universidadabierta.edu.mx/Biblio/B/Badilla-ANALISIS%20Y%20DISENO.html>
- [4] <http://www.edic.lsi.uniovi.es/isoft/recursos/dfd.pdf>
- [5] [http://www.dsic.upv.es/asignaturas/evi/mtp/transpas/PDF/tema9\(pruebas\).pdf](http://www.dsic.upv.es/asignaturas/evi/mtp/transpas/PDF/tema9(pruebas).pdf)
- [6] <http://www.homepage.ufg.edu.sv/~lmartinez/prueba.doc>
- [7] <http://www.homepage.ufg.edu.sv/~lmartinez/mantenim.doc>
- [8] <http://www.acm.org/crossroads/espanol/xrds7-4/onpatrol74.html>