



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

FACULTAD DE INGENIERIA

AUTOMATIZACION DEL PROCESO DE COBRANZA DE UNA EMPRESA HIPOTECARIA.

T E S I S

QUE PARA OBTENER EL TITULO DE: INGENIERO EN COMPUTACION PRESENTAN: JOSEFINA CRUZ ALONSO FERNANDO ESCALANTE MORÁLES ALEJANDRO VILLEGAS OTERO

DIRECTOR DE TESIS: M. EN I. JUAN CARLOS ROA BEIZA



CIUDAD UNIVERSITARIA, D. F.

2002.

TESIS CON FALLA DE ORIGEN



Universidad Nacional  
Autónoma de México



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

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

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

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

PAGINACIÓN

DISCONTINUA

## **AGRADECIMIENTOS**

### **A mis PADRES**

A ellos que han impulsado mis pasos para conseguir mis metas y que han estado a mi lado en este camino apoyándome e impulsándome a continuar y no rendirme, a ellos les agradezco de manera infinita todos sus esfuerzos por brindarme todo, a ellos les dedico este y todos mis trabajos, gracias, lo hemos logrado.

### **A mis AMIGOS**

A todas aquellas personas que me brindaron su apoyo y que han estado conmigo en todo momento para ayudarme cuando lo he necesitado a todos ellos gracias por su amistad y su paciencia.

### **A mi FAMILIA**

Que siempre que he necesitado de su apoyo de cualquier manera me han hecho superar las adversidades gracias a todos ustedes por estar al pendiente de mi.

**A TODOS GRACIAS DE MANERA INFINITA A USTEDES EN CONJUNTO LES PERTENECE ESTE TRABAJO Y LES DEBO LO QUE SOY Y LO QUE SERÉ.**

**ATENTAMENTE  
FERNANDO ESCALANTE MORALES**

**" DIOS, CONCÉDEME  
SERENIDAD PARA ACEPTAR LO QUE NO PUEDE SER CAMBIADO  
VALOR PARA CAMBIAR LO QUE PUEDE SER CAMBIADO, Y  
SABIDURÍA PARA DISCERNIR LO UNO DE LO OTRO "**



## **AGRADECIMIENTOS:**

### **A DIOS**

Porque es Él quien ha guiado mi vida.

### **A mis PADRES**

Porque con su humildad y sencillez me han sabido conducir por el buen camino y me han dado ejemplo de constancia y tenacidad.

### **A mis HERMANOS**

Porque han sido un apoyo en mi camino y un ejemplo a seguir.

### **A mis SOBRINOS y CUÑADOS**

Por su ayuda incondicional y su amistad.

### **A mis AMIGOS**

Porque en cada momento han creído y confiado en mí.

***Gracias a mis compañeros de tesis: Josefina, Fernando, Azucena y Nieves, por su colaboración y apoyo en este trabajo, con el cual nos hemos comprometido a seguir adelante!!!***

**ALEJANDRO VILLEGAS OTERO**

*Dedicada a:*

*Mi Padre, Salvador Cruz E.  
A ti, Pa' que siempre te has esforzado por darme  
todo y siempre lo mejor.  
(tu último error)*

*Mi Madre Josefina Alonso, por su todo su amor,  
por su ardua labor de forjar las bases de mi vida,  
inculcándome siempre el seguir adelante.*

*A todos aquellos que encontraron la manera,  
nunca por el camino fácil,  
de impulsar mi destino.*

*Sus esfuerzos han fructificado!*

*Gracias por su Paciencia, Compresión y Amor  
incondicional a pesar de que no fue fácil.*

*Josefina Cruz Alonso.*

## ÍNDICE

INTRODUCCIÓN	I
CAPÍTULO I ANTECEDENTES .....	
1.1 Panorama general de la empresa	1
1.2 Entorno del departamento de cobranza	9
1.3 Situación actual del proceso de cobranza	13
1.4 Políticas de cobranza de la empresa	17
CAPÍTULO II TEORÍA BÁSICA .....	
2.1 Características, ventajas y desventajas de los lenguajes de programación	22
2.1.1 Tipos de datos y tipificación de datos	25
2.1.2 Subprogramas	26
2.1.3 Estructuras de control	26
2.1.4 Soporte para el enfoque orientado a objetos	26
2.2 Bases de Datos	28
2.2.1 Estructura de datos relacional	29
2.2.2 Manipulación de datos	30
2.2.3 Reglas de integridad	33
2.3 Metodología de Base de Datos y Lenguajes de Programación Orientada a Objetos	37
2.3.1 Metodología de Bases de Datos	37
2.3.2 Metodología Orientada a Objetos	42
2.4 Características, ventajas y desventajas de SQL Server 2000	48
2.5 Características, ventajas y desventajas de PowerBuilder 7.0	57

---

---

**CAPÍTULO III PLANTEAMIENTO DEL PROBLEMA Y PROPUESTA  
DE SOLUCIÓN .....**

<b>3.1</b>	<b>Problemática actual</b>	<b>64</b>
<b>3.2</b>	<b>Requerimientos de la empresa</b>	<b>67</b>
	3.2.1 Estandarización y convenciones de codificación	68
<b>3.3</b>	<b>Búsqueda y análisis de la información</b>	<b>82</b>
<b>3.4</b>	<b>Identificación del problema</b>	<b>92</b>
<b>3.5</b>	<b>Entidades involucradas en el proceso</b>	<b>101</b>
	3.5.1 Intermediario Financiero	101
	3.5.2 Subdirección de Cobranza	102
	3.5.3 Dirección de Finanzas	103
	3.5.4 Institución bancaria	103
<b>3.6</b>	<b>Opciones de solución y elección de la óptima</b>	<b>104</b>
	3.6.1 Captura de datos	105
	3.6.2 Carga de movimientos	107
	3.6.3 Validación y asociación de los pagos recibidos contra el estado de cuenta	107
	3.6.4 Reportes	107
	3.6.5 Mantenimiento a catálogos del sistema	108
	3.6.6 Seguridad	109
	3.6.7 Evaluación de productos para el desarrollo del Back End	109
	3.6.7.1 Descripción de SQL de Oracle 9i	109
	3.6.7.2 Descripción de SQL Server 2000	110
	3.6.7.3 Descripción de MySQL 4.0.1 Sybase	111
	3.6.7.4 Comparación de productos	112
	3.6.8 Evaluación de productos para el desarrollo del Front End	114
	3.6.8.1 Power Builder	115

---

---

3.6.8.2 Delphi	117
3.6.8.3 Visual Basic	119
3.6.8.4 Comparación de productos	120
3.6.9 Elección de la óptima solución	121
3.6.9.1 Elección del Backend	122
3.6.9.2 Elección del Frontend	123

## CAPÍTULO IV DESARROLLO E IMPLANTACIÓN DEL SISTEMA .....

4.1 Aplicación de la metodología elegida (Back – End)	126
4.1.1 Diagrama de contexto	126
4.1.2 Diagrama de flujo	128
4.1.3 Diccionario de datos	134
4.1.4 Diagrama Entidad - Relación	139
4.1.5 Normalización	140
4.2 Construcción del Back – End	143
4.2.1 Diseño del Back – End	143
4.2.2 Construcción del Back – End	147
4.2.3 Seguridad dentro de SQL Server	150
4.2.4 Tamaño de la Base de Datos	152
4.2.5 Seguridad adicional	156
4.3 Diseño y construcción del Front – End	159
4.4 Pruebas y liberación del sistema	171
4.5 Generación de reportes	188
4.6 Factibilidad técnica y operativa	205
4.6.1 Requerimientos mínimos de hardware y software del servidor	205
4.6.2 Requerimientos mínimos de hardware y software de las estaciones de trabajo	206

---

---

4.6.3	Análisis del costo y venta del sistema	210
4.6.4	Implantación del proyecto	215

**CONCLUSIONES**

**APÉNDICE A MANUAL DE USUARIO**

**APÉNDICE B MANUAL TÉCNICO**

**BIBLIOGRAFÍA**

---

## INTRODUCCIÓN

En el presente trabajo se describe el desarrollo de un sistema para automatizar el proceso de cobranza de una empresa dedicada al ramo hipotecario, mismo que actualmente se realiza en forma manual. Ésto con el fin de controlar y agilizar el cobro por cualquier concepto generado de los préstamos que se realizan sobre bienes inmuebles.

En dicha empresa se requiere tener la información actualizada y disponible en todo momento, conocer el estado de la cobranza y contar con la disponibilidad de los recursos activos que posee la empresa.

Actualmente, en la empresa hipotecaria se lleva a cabo la recepción de pagos de forma manual, lo que implica una inversión de tiempo y papeleo considerable e innecesario, además de que existe la posibilidad del traspapeleo de los documentos, cosa que retarda el cobro de los mismos al no tener la información disponible.

Debido al proceso manual no es posible tener con oportunidad la información acerca de los cheques devueltos, ni el motivo por el cual fueron rechazados, por tanto no se puede exigir el pago inmediato.

Todo lo anterior impide llevar un control detallado de quién ha pagado en tiempo, quién en mora y quién no ha cubierto los intereses correspondientes por retardo en el pago o por cheques devueltos.

Viendo la problemática existente en la empresa y las dificultades que ésta tiene para realizar una cobranza sana, se propone un sistema que minimice los errores humanos y optimice el tiempo de recuperación de recursos.

Nuestra propuesta consiste en el desarrollo de un sistema que dé seguimiento a los pagos recibidos, partiendo de la captura de los datos de los documentos, pasando por una validación y asociación de los mismos contra un estado de cuenta electrónico, a fin de identificar el estatus de dichos pagos, lo cual generará reportes que contengan la información relevante para que el usuario tenga un control detallado del proceso de cobranza.

Cabe señalar que este sistema será utilizado por los departamentos de cobranza y finanzas de la empresa, por lo que se requiere implantar filtros de seguridad para acceder a las opciones dentro del sistema.

En el presente trabajo de tesis se expone de manera explícita un panorama general de la empresa y de los departamentos involucrados; una revisión técnica de los temas en los que nos apoyamos para el desarrollo del mismo y toda la documentación necesaria para el análisis, diseño, construcción e implantación del sistema. También se anexan los manuales necesarios para su instalación, uso y mantenimiento.



## **CAPÍTULO I ANTECEDENTES**

### **1.1 Panorama general de la empresa**

La empresa que solicita este sistema dirige sus actividades dentro del ramo hipotecario; se encarga del otorgamiento de créditos (préstamos) para la adquisición de un bien inmueble. Un préstamo hipotecario es aquél que está destinado a facilitar la compra de una casa o departamento, ya sea nuevo o usado.<sup>1</sup>

La utilización de un préstamo hipotecario es una de las vías de financiación para la adquisición de una vivienda más fiable de las que existen hoy en día, además de tener en cuenta el beneficio que se tiene al adquirir un inmueble, que representa una inversión la cual incrementará el valor del patrimonio familiar y evitará el pago de una renta.

Se entiende por vivienda la edificación que constituya la residencia habitual del comprador. Con esta definición se excluyen todas las viviendas movientes como las caravanas, barcos, etc., que no tienen la consideración de edificaciones. La vivienda puede ser nueva, usada o rehabilitada.

La necesidad de un préstamo hipotecario surge por la demanda del vendedor de cobrar todo el dinero de la venta en el momento de la formalización ante Notario de la operación de compraventa, y del comprador por no disponer de la totalidad del dinero en ese momento.

Los esquemas de financiamiento varían de acuerdo a la institución que otorgue el préstamo. En todos los casos, el bien inmueble (casa) queda como garantía hasta el pago del mismo.

---

El préstamo hipotecario es otorgado y concedido por una entidad financiera y de crédito, que en nuestro caso, se otorga a través de intermediarios financieros, los cuales son el puente entre los compradores y la empresa.

En la operación del préstamo hipotecario intervienen el *prestamista* (Entidad financiera y de crédito, esto es, la empresa hipotecaria), que entrega una cantidad (llamada capital) al comprador que se convierte por ello en *prestatario*, para que éste lo devuelva, en el período pactado, más los intereses.

Sin embargo, la generación del préstamo es una operación compleja en cuya tramitación inciden numerosas entidades, que demandan coordinar sus esfuerzos de forma ordenada. Se involucran distintas materias de ámbito económico, tales como contabilidad, cobranza, riesgos, finanzas, análisis de mercados, créditos, tasas, cartera entre otros.

Tratándose del manejo de dinero, y tomando en cuenta el plazo al cual se otorgan los créditos (años), se hace indispensable el correcto desempeño de cada una de las áreas involucradas en el seguimiento que se da al mismo: riesgos y tasas, en el cálculo de las condiciones que se aplicarán; créditos, en el otorgamiento de los mismos; cartera, en el registro de las condiciones otorgadas; contabilidad, en la aplicación de los pagos a las cuentas indicadas; cobranza, en el seguimiento de los pagos dentro de los plazos establecidos.

Ésta última es un área crítica, pues es la encargada de recuperar los recursos de la institución para que ésta pueda contar con liquidez y tenga la posibilidad de generar más recursos.

La recuperación de dichos recursos se realiza recibiendo los pagos por el crédito otorgado, que regularmente se financia en mensualidades. Estos pagos mensuales pueden ser fijos o variables, dependiendo de las condiciones pactadas en el

---

<sup>1</sup> <http://www.generalhipotecaria.com>

---

financiamiento. Normalmente, el monto del pago incluye una parte de pago a capital<sup>2</sup>, y otra parte incluye intereses sobre el monto del crédito

Las cualidades que debe tener un préstamo hipotecario son, en forma detallada, las siguientes:

- **Importe máximo del préstamo:** El monto del importe que una entidad puede facilitar, varía en función de diversos factores:
  - El primer límite viene determinado por el **valor de la casa**, según se desprende del avalúo realizado por una empresa autorizada, lo que permite al prestatario verificar si la cantidad que le pide el vendedor se ajusta a los valores de mercado.

El avalúo de la vivienda es, por tanto, otra forma de aumentar tanto la seguridad jurídica como la de la entidad de crédito, el comprador puede incluso encomendar la operación de valuar el inmueble de forma independiente a la empresa hipotecaria. El valor que se le asigne a la casa por el citado avalúo, no tiene por qué coincidir con el precio que se haya pagado por el inmueble.

Las empresas hipotecarias, de acuerdo con la legislación vigente, pueden prestar entre el 70 ó el 80 % del valor de avalúo de la vivienda, aunque en muchos casos llega incluso hasta el 100% y, en otros incluso más, para cubrir los gastos e impuestos, pudiendo complementarse el préstamo hipotecario con préstamos personales. El límite del 80 % es, en general, el que rige los préstamos para la construcción, rehabilitación o adquisición de todo tipo de viviendas, y no sólo las de protección oficial.

Esta limitación tiene por finalidad conseguir que los créditos hipotecarios puedan ser liquidados por entero, en caso de falta de pago, pese a las oscilaciones que en valor del inmueble pudiese producirse en un futuro o pese a los numerosos

---

<sup>2</sup> El pago recibido descuenta directamente parte del monto de la deuda.

gastos que las entidades prestamistas tienen en una ejecución hipotecaria como por ejemplo intereses, costos por conceptos de cobranza o legales, entre otros.

- La segunda limitación está marcada por **los ingresos**. Como regla general, lo habitual es que la cuota mensual del préstamo hipotecario no sobrepase el 30 % ó 40 % de los ingresos netos mensuales, que el comprador perciba. Tanto para el prestatario como para el prestamista resulta más seguro, ya que se evita correr riesgos de impago en el momento en que surja algún gasto excepcional en su presupuesto.
- **Plazo del préstamo:** el plazo del préstamo puede ser condicionante, ya que un plazo corto implica cuotas más elevadas. Un plazo mayor, convierte las cargas del préstamo en más accesible.

En general, con intereses bajos, el alargamiento del plazo produce una mayor disminución de la cuota, mientras que con tipos de interés altos, el plazo más largo no produce un descenso significativo de la cuota a pagar.

Las empresas de crédito suelen conceder sus préstamos hipotecarios a un plazo de 10, 15, 20, 25 ó 30 años.

- **La amortización:** La amortización es el pago del capital pendiente. En los préstamos con sistema de amortización francés, al principio se pagan más intereses y se amortiza menos capital. Lo contrario que sucede al final de la vida del préstamo, donde se amortiza más capital y son menos los intereses.

Por tanto, en la primera fase de vigencia del préstamo hipotecario se pagan principalmente intereses. La amortización del capital va siendo mayor, a medida que avanza la vida del préstamo.

---

Existe la posibilidad de amortizar anticipadamente el préstamo hipotecario, ya sea en su totalidad o en parte. Normalmente, ello implica una penalización, pero supone una reducción en el monto de los intereses a pagar. Si el préstamo es a tipo de interés variable, la comisión por cancelación anticipada no puede ser, por Ley, superior al 1% del capital que se amortice.

- **La cuota:** Es el importe que se compromete a pagar periódicamente quien suscribe un préstamo hipotecario a la empresa hipotecaria que se lo ha concedido. La cuota puede ser mensual, trimestral, semestral o anual. Lo normal es que sea mensual. Como ya hemos comentado la cuota incluye el pago de los intereses y la amortización del capital.

Asimismo, la cuota puede ser *constante*, es decir, sin variación en su importe, durante toda la vida del préstamo, lo que ocurre con los préstamos a tipo fijo y con los préstamos con cuota constante (donde se juega con el tiempo, para acoplar el capital pendiente al tipo de interés actual) o puede ser *variable* (préstamos variables) en función de la modificación del tipo de interés.

- **La carencia:** Cuando hablamos de hipotecas se refiere al período en el que no se abona la parte de amortización de capital y sólo se pagan los intereses de ese capital (carencia de capital).

También podemos hablar de carencia de amortización y de intereses, cuando los intereses calculados tampoco se pagan y por tanto, incrementa la deuda del cliente para el nuevo cálculo de intereses (llamada carencia total). Este sistema es muy poco utilizado en la actualidad en las hipotecas.

Una de las cuestiones más importantes es elegir el tipo de préstamo de entre estas dos opciones: tipo de interés fijo o variable. En el fondo, esta elección supone una apuesta por la evolución del precio del dinero en el mercado, y claro está, la apuesta depende

de una multitud de factores, que escapan, a las previsiones más exactas: factores políticos, económicos, la evolución de Europa, etc.

Lógicamente, el tipo de interés es un aspecto muy importante del préstamo ya que, unido al plazo, determina lo que el prestatario pagará a lo largo de los años y de la vida del préstamo hipotecario. Sin embargo, no hay que ver el tipo de interés como un elemento aislado. Además de valorar aspectos menos tangibles como la relación con su entidad, debe prestar atención a todos los demás elementos del préstamo: la modalidad del tipo de interés (fijo o variable), las comisiones que se aplican, las cuotas a pagar y su periodicidad y el plazo de amortización.

Actualmente, la oferta de préstamos hipotecarios es muy amplia y variada. Pese a la multitud de nombres comerciales diferentes que existen, se pueden distinguir los siguientes tipos de préstamos hipotecarios:

- *Préstamo hipotecario a tipo fijo:* Mantendrá el mismo tipo de interés, y por lo tanto, la cuota será invariable a lo largo de toda la vida del préstamo, o sea que el comprador pagará todos los meses lo mismo, eso le dará cierta seguridad (si los tipos de interés suben, no se verá afectado), pero también tiene un inconveniente (si los tipos bajan no podrá beneficiarse); suelen tener un plazo más ajustado de duración (alrededor de 12 años) y la comisión por amortización o cancelación anticipada es mayor, porque, al contrario de lo que ocurre con los préstamos hipotecarios a tipo variable, no se fija ni se estipula una comisión de cancelación máxima, reduciéndose así, en algunas ocasiones, la rentabilidad financiero-fiscal del préstamo.
- *Préstamo hipotecario a interés variable:* Varía en el tiempo, normalmente cada año o cada seis meses. Esta clase de préstamos, sí permite beneficiarse de las rebajas de los tipos aunque, lógicamente, también, se afecta por las subidas. Su principal ventaja es que no contemplan el riesgo de que el interés del préstamo se quede

de una multitud de factores, que escapan, a las previsiones más exactas: factores políticos, económicos, la evolución de Europa, etc.

Lógicamente, el tipo de interés es un aspecto muy importante del préstamo ya que, unido al plazo, determina lo que el prestatario pagará a lo largo de los años y de la vida del préstamo hipotecario. Sin embargo, no hay que ver el tipo de interés como un elemento aislado. Además de valorar aspectos menos tangibles como la relación con su entidad, debe prestar atención a todos los demás elementos del préstamo: la modalidad del tipo de interés (fijo o variable), las comisiones que se aplican, las cuotas a pagar y su periodicidad y el plazo de amortización.

Actualmente, la oferta de préstamos hipotecarios es muy amplia y variada. Pese a la multitud de nombres comerciales diferentes que existen, se pueden distinguir los siguientes tipos de préstamos hipotecarios:

- *Préstamo hipotecario a tipo fijo:* Mantendrá el mismo tipo de interés, y por lo tanto, la cuota será invariable a lo largo de toda la vida del préstamo, o sea que el comprador pagará todos los meses lo mismo, eso le dará cierta seguridad (si los tipos de interés suben, no se verá afectado), pero también tiene un inconveniente (si los tipos bajan no podrá beneficiarse); suelen tener un plazo más ajustado de duración (alrededor de 12 años) y la comisión por amortización o cancelación anticipada es mayor, porque, al contrario de lo que ocurre con los préstamos hipotecarios a tipo variable, no se fija ni se estipula una comisión de cancelación máxima, reduciéndose así, en algunas ocasiones, la rentabilidad financiero-fiscal del préstamo.
- *Préstamo hipotecario a interés variable:* Varía en el tiempo, normalmente cada año o cada seis meses. Esta clase de préstamos, sí permite beneficiarse de las rebajas de los tipos aunque, lógicamente, también, se afecta por las subidas. Su principal ventaja es que no contemplan el riesgo de que el interés del préstamo se quede

desfasado, porque periódicamente (normalmente cada 12 meses) se actualiza a los precios de mercado.

Las principales características de este tipo de préstamos es que el interés varía a ritmo de mercado, el plazo de amortización es mayor (puede llegar a ser de 20 ó 30 años) y la comisión por cancelación o amortización anticipada no supera el uno por ciento.

- *Préstamo hipotecario a interés mixto:* Combinan un período en el que el interés permanece fijo (2, 3 ó más años) y otro en el que éste es variable y se va ajustando al mercado. El plazo de amortización y las comisiones por cancelación anticipada suelen ser parecidos a los variables. Combinan las ventajas e inconvenientes de los préstamos hipotecarios a tipo fijo y los préstamos hipotecarios a interés variable aunque, al acotar dos períodos diferentes, los riesgos quedan más diluidos. Por lo demás, su funcionamiento es exactamente igual que los préstamos a interés variable.
- *Préstamo hipotecario de cuota fija:* Son préstamos a interés variable, pero con cuota de amortización fija, por lo que se asemejan bastante a los préstamos a tipo fijo en la medida en que el cliente paga siempre la misma cuota con independencia y al margen de la evolución de los tipos de interés.

La diferencia está en que si los tipos de interés suben, es decir, hay una revisión al alza, en lugar de pagar más cuota, se alarga el plazo de amortización del préstamo; y si los tipos de interés bajan o existe una revisión a la baja, el plazo de amortización se acorta.

Lógicamente, su principal problema reside en la incertidumbre que creará al prestatario, ya que no se sabe con certeza cuándo terminará la vigencia del préstamo. Por el contrario, tiene la ventaja, de que la cuota a abonar al



intermediario financiero no variará ni un céntimo a lo largo de toda la vida del préstamo.

## 1.2 Entorno del departamento de cobranza

La importancia que tienen las ventas a crédito en cualquier empresa radica en que si no se da crédito disminuye en gran proporción el nivel de ventas, pero lo más relevante de dar crédito consiste en administrar correctamente los cobros para evitar quedarse sin liquidez. La entidad a quien corresponde dicha responsabilidad dentro de la empresa es el área de Cobranza, quien se encarga de solicitar a los deudores la liquidación de sus créditos, no sólo de los intereses generados, sino también de exigir los pagos por multas, anticipos y más. Sus actividades involucran financiar la inversión de la empresa en cuentas por cobrar, hacer cumplir las políticas de crédito establecidas, investigar a los clientes potenciales y vigilar la cobranza.

Los créditos que se otorgan a los clientes al concederles un tiempo razonable para que paguen los servicios adquiridos después de haberlos recibido, son lo que se conoce como *cuentas por cobrar*. El término por cobrar significa la promesa del cliente de pagar en una fecha futura, con dinero, el importe que le fue cargado por adquirir el bien.

Los préstamos a plazo se amortizan generalmente mediante varios pagos de igual cuantía a lo largo del período de préstamo, aunque a veces puede haber un pago final importante o simplemente un único pago al vencimiento. Esto es lo que se define como Cobranza: el seguimiento de dichos pagos a lo largo de la vida del crédito, lo cual puede resultar costoso.

La mayoría de las empresas consideran las cuentas por cobrar como un instrumento de mercadotecnia para promover las ventas y las utilidades. El objetivo de la política de cuentas por cobrar que siguen las empresas, por lo general, consiste en fomentar las ventas y ganar más clientes otorgándoles crédito.<sup>3</sup>

---

<sup>3</sup> Bolton, Steven E., "Administración Financiera", 1983

## 1.2 Entorno del departamento de cobranza

La importancia que tienen las ventas a crédito en cualquier empresa radica en que si no se da crédito disminuye en gran proporción el nivel de ventas, pero lo más relevante de dar crédito consiste en administrar correctamente los cobros para evitar quedarse sin liquidez. La entidad a quien corresponde dicha responsabilidad dentro de la empresa es el área de Cobranza, quien se encarga de solicitar a los deudores la liquidación de sus créditos, no sólo de los intereses generados, sino también de exigir los pagos por multas, anticipos y más. Sus actividades involucran financiar la inversión de la empresa en cuentas por cobrar, hacer cumplir las políticas de crédito establecidas, investigar a los clientes potenciales y vigilar la cobranza.

Los créditos que se otorgan a los clientes al concederles un tiempo razonable para que paguen los servicios adquiridos después de haberlos recibido, son lo que se conoce como *cuentas por cobrar*. El término por cobrar significa la promesa del cliente de pagar en una fecha futura, con dinero, el importe que le fue cargado por adquirir el bien.

Los préstamos a plazo se amortizan generalmente mediante varios pagos de igual cuantía a lo largo del período de préstamo, aunque a veces puede haber un pago final importante o simplemente un único pago al vencimiento. Esto es lo que se define como Cobranza: el seguimiento de dichos pagos a lo largo de la vida del crédito, lo cual puede resultar costoso.

La mayoría de las empresas consideran las cuentas por cobrar como un instrumento de mercadotecnia para promover las ventas y las utilidades. El objetivo de la política de cuentas por cobrar que siguen las empresas, por lo general, consiste en fomentar las ventas y ganar más clientes otorgándoles crédito.<sup>3</sup>

---

<sup>3</sup> Bolten, Steven E., "Administración Financiera", 1983

---

Las cuentas por cobrar o cobranza, inmovilizan moneda corriente, que de estar en disposición inmediata podría utilizarse o invertirse en activos adicionales que generen ganancias. Son un costo necesario para la empresa, ya que el ofrecimiento de crédito a los clientes permite que la empresa alcance un nivel más alto de ventas.

Los créditos garantizados con cuentas por cobrar no sólo pueden obtenerse de los bancos comerciales, sino también de sociedades de financiación, las cuales están especializadas por rubros, como es el caso de los Intermediarios Financieros que trabajan con la empresa hipotecaria.

Las cuentas por cobrar forman parte del activo circulante. Entre las principales cuentas que están incluidas en este rubro son:

- Clientes
- Documentos por cobrar
- Funcionarios y empleados
- Deudores diversos

Pueden existir algunas otras cuentas como "intereses por cobrar", "dividendos por cobrar", "rentas por cobrar", aunque de todos estos tipos, las dos más conocidas son los documentos por cobrar y los clientes.

El tipo de cuentas por cobrar que a nuestra problemática atañe es el que involucra los documentos por cobrar, pues los intermediarios financieros realizan sus pagos a través de documentos bancarios como cheques o depósitos.

El objetivo de la política de cuentas por cobrar debe enfocarse a maximizar las utilidades y el rendimiento de la inversión y que no se abuse del procedimiento hasta el grado en que el otorgamiento de créditos deja de ser rentable y atractivo en términos de rendimientos sobre inversión. Se deberá evitar otorgar crédito a clientes potenciales que probablemente tendrán dificultades para pagar o que inmovilizarán los recursos de la empresa demorando el pago por largo tiempo (para evitar este riesgo es necesario realizar un estudio socioeconómico con el fin de evaluar la capacidad de pago del

interesado). No conviene financiar a clientes que ofrecen demasiado riesgo o que requieren un total de cuentas por cobrar tan considerable que la inversión sólo sea rentable marginalmente porque absorbe recursos que se podrían destinar a financiar otros proyectos más atractivos.

Los factores de costo y beneficio que se deben tomar en cuenta para establecer las políticas de cuentas por cobrar son:

### Costos

- **De cobranza:** comprende el sostener y operar de modo general un departamento de crédito (contadores, ayudantes, actuarios) proveyéndolo de elementos esenciales como papelería, gastos de correo, teléfonos y computadoras.
- **De capital:** Una vez que la empresa decide otorgar crédito, tiene que obtener los recursos para financiarlo. Debe pagar a sus empleados mientras espera que el cliente pague por él mismo. La diferencia de tiempo implica que tendrá que conseguir recursos por fuera (o utilizar los generados internamente, que se podrían invertir con más provecho en otra parte) para cumplir con sus pagos mientras espera los de sus deudores. El capital es caro y la empresa tiene que soportar el costo.
- **De morosidad:** Suceden cuando el cliente se demora en pagar. El hecho de que no pague en tiempo aumenta otros costos que van asociados con la cobranza normal. Habrá que pagar el costo de avisos, demandas legales, llamadas telefónicas y otros más que se emplean para poder realizar el cobro. La morosidad en los pagos inmoviliza recursos que podrían estar generando beneficios en otra parte, lo cual crea un costo de oportunidad por el tiempo adicional en que los recursos están inmovilizados después de transcurrido el período de cobro normal.
- **De incumplimiento:** Ocurren cuando el deudor no paga definitivamente. Además de los costos de cobranza, de capital y de morosidad en que ocurrió hasta ese momento, la empresa pierde el costo de los artículos vendidos y no pagados.

Tendrá que cancelar la venta en cuanto decida que la cuenta morosa se ha vuelto incobrable.

### **Beneficios**

La empresa obtiene de su política de cuentas por cobrar beneficios que se deben comparar con los costos para determinar la conveniencia de una política determinada. Los beneficios consisten en el aumento de las ventas y las utilidades que se esperan de una política más liberal.

No conviene extender demasiado crédito a un cliente, para evitar que la empresa reciba un duro golpe en el caso de que dicho cliente falle inesperadamente. Hay que fijar un límite para el crédito que será concedido, aunque la probabilidad de que no cumpla se considere muy remota. A los clientes más confiables se les podrá conceder más crédito que a los que lo son menos.

Es muy importante vigilar constantemente las cuentas por cobrar, para detectar cualquier indicación de deterioro en la capacidad de pago de algún cliente o de que se esté concediendo un crédito excesivo.

### **1.3 Situación actual del proceso de cobranza**

Actualmente el proceso de cobranza, dentro de la empresa, se lleva a cabo de forma manual, se reciben los cheques por diversos conceptos acompañados de un documento en el cual se especifica el concepto del pago, el número del cheque, el monto y se revisa que venga firmado, además se checa que la firma se encuentre registrada en el catálogo de la empresa.

En primera instancia, los cheques son sometidos a una inspección visual, en la cual se valida que la fecha no sea mayor al día en que el pago es recibido; que la cantidad indicada con letra y número coincidan y, que a su vez, correspondan con el monto que se indica en el documento anexo y que ambos (tanto cheque como carta) exhiban una firma autorizada. Aprobados todos estos conceptos, el cheque es recibido sin importar el monto indicado.

Una vez que los documentos han sido revisados satisfactoriamente, el paso siguiente es validar, contra un reporte, la cantidad que debió ser pagada. En caso de que el monto del pago sea mayor al esperado, se procede a dar aviso al Intermediario Financiero, vía telefónica, informándole que le será devuelta la diferencia, en un plazo no mayor a 3 días hábiles, mediante un depósito directo en su cuenta bancaria. Si el monto del pago recibido es menor, la diferencia se deposita contablemente en cuentas de deudor y se exige el pago de forma inmediata, vía telefónica, y de ser posible, se cobran intereses moratorios.

Al final del día todos los pagos son recopilados para elaborar una relación que incluye: número de cheque, institución bancaria del mismo y monto. Concluido el proceso, este inventario de pagos se remite al área de Finanzas.

Hasta aquí llega la responsabilidad del área de cobranza en el seguimiento de los pagos.

Es ahora responsabilidad del área de Finanzas generar las fichas de depósito correspondientes a los documentos para enviarlas (al día siguiente, a primera hora) a depósito bancario.

Los cheques pertenecientes a la misma institución donde son depositados, se abonan de forma inmediata en la cuenta de la empresa hipotecaria, previa comprobación de la disposición de fondos en la cuenta del Intermediario Financiero emisor del pago. En tanto que aquellos que pertenecen a otra institución bancaria, son abonados salvo buen cobro y es hasta el día siguiente cuando se puede disponer del monto depositado.

Es ahora el área de Finanzas quien se encarga de verificar, mediante el uso de una línea empresarial establecida con el banco, los cheques que fueron cobrados y los que fueron rechazados.

El organigrama de las áreas involucradas en el proceso de la cobranza puede verse en la figura 1.3.1.

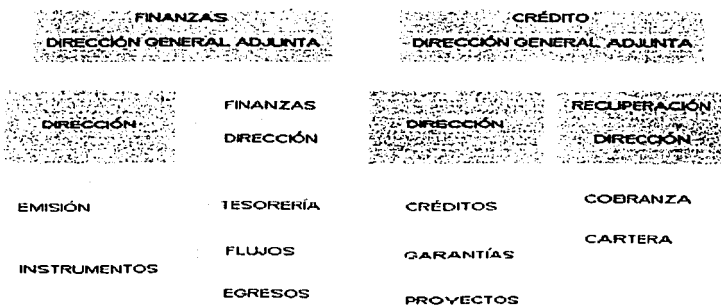


Figura 1.3.1 Organigrama del proceso de cobranza



La línea empresarial (Fig. 1.3.2) es un enlace dedicado<sup>4</sup>, mediante el cual es posible tener acceso a la Base de Datos de una empresa en forma remota, para realizar consultas e inclusive registrar información.

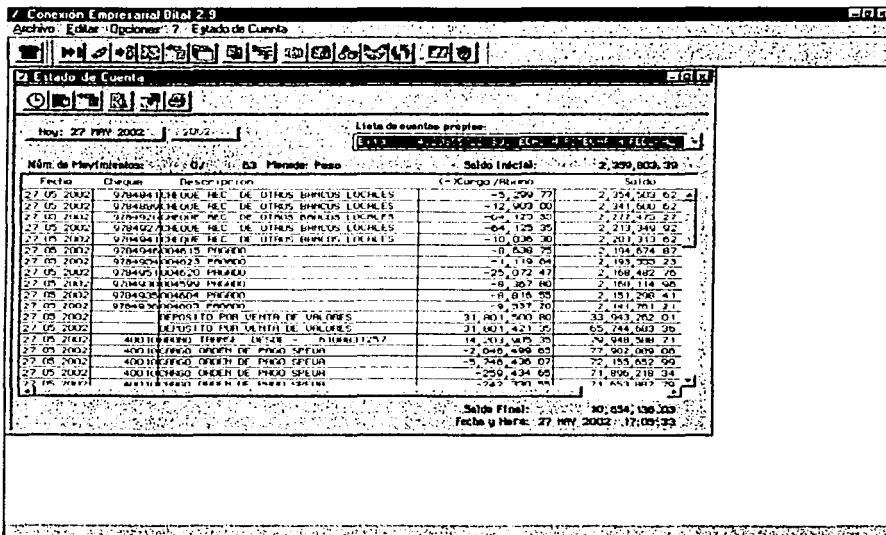


Figura 1.3.2 Línea Empresarial

Para tal efecto se tiene una computadora de consulta, que permite generar un archivo en Excel con el cual se revisan los estados de cuenta de la empresa. En la figura 1.3.3 se muestra un archivo generado a partir de la línea empresarial, con el formato de las columnas que es posible exportar. En dicho archivo se listan todos los movimientos registrados (no sólo pagos), por lo que es necesario identificar visualmente los correspondientes a la cobranza, según los documentos depositados.

<sup>4</sup> Término empleado en el ámbito de cómputo para indicar que se tiene una línea telefónica que es destinada solamente para realizar una conexión directa entre lugares lejanos.

CARGOS		C		D	E	F	G	H
2	17 MAY 2002	00000		3,702.62		2,978,271.14	0109	0
3	17 MAY 2002	00463.4		10,070.29	0.00	3,079,341.43	0109	0
4	17 MAY 2002	00469.6		10,070.87	0.00	3,189,412.30	0109	0
5	17 MAY 2002	00476.7		4,011.39	0.00	3,233,423.69	0109	0
6	17 MAY 2002	00482.8		2,265.57	0.00	3,256,179.26	0109	0
7	17 MAY 2002	0		0.00	3,444.13	3,257,623.39	0648	0
8	17 MAY 2002	0		0.00	19,714,291.17	3,277,314.56	0648	0
9	17 MAY 2002	0		0.00	97,12,361.01	3,374,675.57	0648	0
10	17 MAY 2002	00010	CARGO TRANSF A - 4007599178	541,406.15	0.00	3,916,081.72	0649	06643
11	17 MAY 2002	00010	CARGO TRANSF A - 4001137103	6,11974.72	0.00	4,527,056.44	0649	06644
12	17 MAY 2002	00010	CARGO TRANSF A - 4005638483	1,028,836.45	0.00	5,555,892.89	0649	06646
13	17 MAY 2002	00010	CARGO TRANSF A - 401490725	727,073.93	0.00	6,282,966.82	0649	06640
14	17 MAY 2002	00010	CARGO TRANSF A - 4015117922	1,051,923.31	0.00	7,334,890.13	0649	06651
15	17 MAY 2002	00010	CARGO TRANSF A - 4018916421	4,599,726.47	0.00	11,934,616.60	0649	06653
16	17 MAY 2002	00010	CARGO ORDEN DE PAGO SPEUA	3,705,12.07	0.00	15,639,738.67	0405	0
17	17 MAY 2002	00010	CARGO ORDEN DE PAGO SPEUA	7,449,882.86	0.00	23,089,621.53	0405	0
18	17 MAY 2002	00010	CARGO ORDEN DE PAGO SPEUA	5,682,62.97	0.00	28,772,254.50	0405	0
19	17 MAY 2002	00010	CARGO ORDEN DE PAGO SPEUA	2,178,180.53	0.00	30,950,435.03	0405	0
20	17 MAY 2002	00010	CARGO ORDEN DE PAGO SPEUA	9,969,804	0.00	40,920,239.03	0405	0
21	17 MAY 2002	0	DEPOSITO CON DOCUMENTOS	0.00	1,117.34	41,031,356.37	0103	0
22	17 MAY 2002	0	DEPOSITO CON DOCUMENTOS	0.00	2,664,41.67	43,695,768.04	0103	0
23	17 MAY 2002	0	DEPOSITO CON DOCUMENTOS	0.00	1,652,407.21	45,348,175.25	0103	0
24	17 MAY 2002	0623416	DEPOSITO CHEQUE BCC042 C7A0000055514 T1	0.00	10,046,184	55,394,359.29	0109	0
25	17 MAY 2002	0607	DEPOSITO CHEQUE BCC017 C7A0000011649 T1	0.00	10,999,36.43	66,393,725.72	0109	0
26	17 MAY 2002	0623416	DEPOSITO CHEQUE BCC042 C7A0000055514 T1	0.00	10,492,384	76,886,110.12	0109	0
27	17 MAY 2002	0607	DEPOSITO CHEQUE BCC017 C7A0000011649 T1	0.00	10,999,36.44	87,885,476.56	0109	0
28	17 MAY 2002	0623416	DEPOSITO CHEQUE BCC042 C7A0000055514 T1	0.00	10,456,63.84	98,342,110.40	0109	0
29	17 MAY 2002	0607	DEPOSITO CHEQUE BCC017 C7A0000011649 T1	0.00	10,999,36.43	109,341,476.83	0109	0
30	17 MAY 2002	0608	DEPOSITO CHEQUE BCC017 C7A0000011649 T1	0.00	2,117,873.70	111,459,350.53	0109	0
31	17 MAY 2002	7,59	IMPUESTO CHEQUE BCC016 C7A0000055514 T1	0.00	9,662.00	121,121,012.53	0109	0
32	17 MAY 2002	06010	PAGO SIMPHE ED1011H1H1 C010003300005179	1,348,372.00	0.00	122,469,384.53	0609	33007
33	17 MAY 2002	01007	DEPOSITO CHEQUE BCC012 C7A0000055514 T1	0.00	14,191.88	136,660,576.41	0109	0
34	17 MAY 2002	0	17:ARGO PAGO DE SERVICIOS	5,189.10	0.00	141,849,765.51	01192	0
35	17 MAY 2002	00009	ABONO TRANSF DESDE - 4007806748	0.00	34,40.46	141,884,165.97	06423	16462
36	17 MAY 2002	00001	ABONO TRANSF DESDE - 4015675764	0.00	12,338.02	154,222,104.00	06423	70381
37	17 MAY 2002	04331	SPEUA DE SERFIN PATRIMONIO O SA	0.00	264,300.94	154,486,404.94	06061	0
38	17 MAY 2002	0048879	00488 DEPOSITADO	0.00	1,623,272.17	156,109,677.11	06363	0

Figura 1.3.3 Archivo generado en la línea empresarial

A pesar de que es posible verificar los depósitos inmediatamente después de haber realizado los mismos, sólo se debe tener en cuenta que los documentos recibidos salvo buen cobro no podrán ser dispuestos el mismo día. Y he aquí un inconveniente, puesto que en la línea empresarial aparecen como abonos y en caso de ser rechazados aparecerán (al día siguiente) como cargos.

#### 1.4 Políticas de cobranza de la empresa

Las condiciones de pago que ofrece la empresa se imponen de acuerdo al campo en el cual opere y tienen relación directa con:

- *Las normas de cobro:* Estipuladas por el reglamento interno de la empresa, para mantener su posición competitiva.
- *El estándar de políticas de crédito:* Es el criterio para determinar a quien debe concedérsele el crédito.
- *Las políticas de cobro:* Determinan las gestiones que tiene que hacer la empresa para cobrar oportunamente las cuentas por cobrar<sup>5</sup>.

Todos los cambios que se presenten en estos factores pueden determinar el plazo promedio del periodo de cobro sosteniendo o aumentando las utilidades totales.

Las políticas de cobro son determinadas por cada empresa, aunque existen algunas con denominadores en común, de las cuales podemos citar tres:

- *Venta al contado:* No otorga crédito comercial, no habrá costos de cobranza ni de capital, puesto que no se ha invertido dinero para financiar las cuentas por cobrar. Es obvio que tampoco habrá costos de morosidad e incumplimiento, de manera que el costo total de la política es cero, como puede apreciarse en la figura 1.4.1.
- *Neto a 30 días:* Si la empresa concede crédito a 30 días, incurre en dos tipos de costos: el del capital necesario para financiar las cuentas por cobrar durante 30 días y las de cobranza (contabilidad, sueldos de administración de créditos). En la figura 1.4.1 puede verse que los costos de capital aumentan mientras más tiempo permanezca sin cobrar la cuenta, porque la empresa debe pagar intereses por cada día transcurrido. Los costos de cobranza permanecen como un costo constante sumado a los costos de capital en el curso de los primeros 30 días en que la cuenta

<sup>5</sup> <http://www.gestlopolls.com/recursos/experto/cats/exp/pagans/fin/no%2013/creditesycobranza.htm>

permanece sin cobrar. Para compensar los mayores costos de cobranza están los incrementos que se esperan en las ventas y en las utilidades.

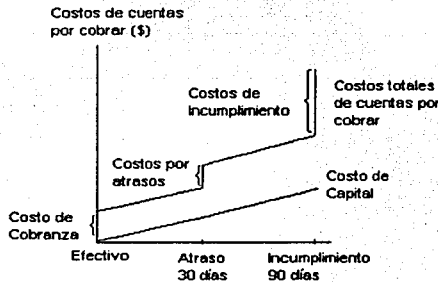


Figura 1.4.1 Costos de cuentas por cobrar, en relación con la política adoptada

- *2/10, neto a 30 días*: En este caso el cliente puede hacer una de dos cosas: comprar los servicios y no aprovechar algún descuento por pronto pago, o comprar los servicios aprovechando los descuentos. La primera opción será igual que si estuvieran sujetos a la política de 30 días. No hay incentivo para comprar a la empresa más de lo que se le compraba con dicha política. La segunda opción, atraerá más clientes y aumentará las ventas previstas. Claro que es necesario examinar los beneficios y costos adicionales asociados con el descuento, para determinar la utilidad de esta política.

En el caso de los créditos otorgados por la empresa hipotecaria para la adquisición de las viviendas, se establecen políticas de cobro en plazos de 20 ó 30 años. Dichas políticas contemplan el caso de incumplimiento aplicando una tasa de intereses moratorios sobre el monto del pago esperado, la cual estipula aplicar 3 veces la tasa de CETES<sup>6</sup> de la subasta vigente al día en que se realiza el pago, por el número de días de retraso.

<sup>6</sup> Certificados de la Tesorería de la Federación, emitidos por Banco de México.

Los Certificados de la Tesorería de la Federación (CETES) son el instrumento de deuda bursátil más antiguo emitido por el Gobierno Federal. Se emitieron por primera vez en enero de 1978 y desde entonces constituyen un pilar fundamental en el desarrollo del mercado de dinero en México. Estos títulos pertenecen a la familia de los bonos cupón cero, esto es, se comercializan a descuento (por debajo de su valor nominal), no devengan intereses en el transcurso de su vida y liquidan su valor nominal en la fecha de vencimiento.

Se pueden emitir a cualquier plazo, siempre y cuando su fecha de vencimiento coincida con un jueves o la fecha que sustituya a éste en caso de que fuera inhábil. De hecho, estos títulos se han llegado a emitir a plazos mínimos de 7 días y a plazos máximos de 728 días. En la actualidad se emiten y colocan a plazos de 28 días y 91 días, y a plazos cercanos a los seis meses y un año.

Estos títulos no devengan intereses debido a que son bonos cupón cero. Sin embargo, la tasa de interés del título está implícita en la relación que existe entre su precio de adquisición, el valor nominal del título y su plazo de vencimiento.

La colocación de estos títulos se realiza mediante subasta, en la cual los participantes presentan posturas por el monto que desean adquirir y la tasa de descuento que están dispuestos a pagar. La convención actual del mercado es cotizarlos a través de su tasa de rendimiento.

La clave de identificación de la emisión de los CETES está compuesta por ocho caracteres, el primero para identificar el título ("B"), el segundo es un espacio en blanco, y los seis restantes para indicar su fecha de vencimiento (año, mes, día). Como se puede observar, lo relevante para identificar un CETE es su fecha de vencimiento, esto significa que dos CETES emitidos en fechas distintas pero que vencen el mismo día, cuentan con la misma clave de identificación, por lo que son indistinguibles entre sí.

Dicha tasa por incumplimiento se cobra al Intermediario Financiero, pues es éste quien debe gestionar el proceso de la cobranza ante el acreditado.



## **CAPÍTULO II    TEORÍA BÁSICA**

### **2.1    Características, ventajas y desventajas de los lenguajes de programación**

Los lenguajes de programación son la manera en la que nos comunicamos con las computadoras. Podemos señalar que este proceso de codificación en sí es una actividad humana. Es por ello que los aspectos psicológicos del lenguaje afectan de manera directa la calidad de la comunicación entre el usuario y la computadora. Las características de ingeniería de un lenguaje de programación tienen un impacto importante sobre el éxito de un proyecto de desarrollo de software. Finalmente, las características técnicas de un lenguaje influyen de manera importante la calidad del diseño. Por tal motivo, las características técnicas afectan los aspectos humanos como a los de la ingeniería del software.

Sabemos de antemano que el proceso de codificación traduce una representación para el software, dada por un diseño detallado, a una realización en un lenguaje de programación. Este proceso continúa cuando un compilador acepta el código fuente como entrada y produce como salida un código objeto dependiente de la máquina. Más tarde, la salida del compilador se traduce a código de máquina. —las instrucciones reales, las cuales dirigen la lógica microprogramada de la unidad central de proceso.

El primer paso en la traducción, es un punto fundamental dentro de la programación, en el proceso de traducción puede aparecer “ruido” de muchas formas distintas, es decir, pueden existir ciertas interrupciones por distintas causas, como el omitir algún carácter lo cual impide que se compile de manera correcta algún programa, o también puede ser que el nombre de los apuntadores difiera y por tanto no exista relación alguna, esto solo por mencionar algunos aspectos. La interpretación equivocada de las especificaciones del diseño detallado puede conducir a un código fuente erróneo. También podemos mencionar que la complejidad o las restricciones de los lenguajes pueden conducir a un código fuente que resulte difícil de probar y mantener.



---

Las características básicas de los lenguajes de programación deben ser centradas principalmente en las necesidades del proyecto que se va a desarrollar, esto es debido a que se tiene que analizar perfectamente el lenguaje que se apegue y que sea de mayor utilidad para el proyecto en cuestión. Las características básicas desde un punto de vista de la ingeniería para un lenguaje de programación, podrían ser las siguientes:

- Facilidad de traducción del diseño previamente realizado, al código.
- Contar con un compilador eficiente.
- Portabilidad del código fuente.
- Disponibilidad de herramientas de desarrollo.
- Facilidad de mantenimiento.

El paso de codificación inicia una vez que se han definido, revisado y modificado, si fuera el caso, el diseño previamente realizado. El grado de facilidad de la traducción del diseño que es la primera característica que se deberían de tomar en cuenta, proporciona una indicación de cómo se aproxima un lenguaje de programación a la representación del diseño. Como ya sabemos, un lenguaje que implante de forma directa las construcciones estructuradas, que incluya estructuras de datos sofisticadas, una E/S especializada, posibilidades de manipulación de bits y construcciones orientadas a objetos, hará que la traducción del diseño del código fuente sea mucho más fácil.

Por otro lado, la portabilidad del código fuente la podemos interpretar de distintas maneras, como por ejemplo, el código fuente permanece inalterado cuando cambia su entorno de funcionamiento, que éste puede ser transportado de un procesador a otro y de un compilador a otro sin ninguna o muy pocas modificaciones, y por último, que el código puede ser integrado en diferentes paquetes de software sin que prácticamente se requieran modificaciones y en su caso sean mínimas, debidas a las características propias del lenguaje de programación.

---

De las características mencionadas con antelación acerca de la portabilidad, la segunda es, la más frecuente. La estandarización (por la ISO –Organización Internacional de Estándares- y/o el ANSI –Instituto Nacional Americano de Estándares) continúa siendo el principal esfuerzo para la mejora de la portabilidad de los lenguajes de programación. Desafortunadamente, la mayoría de los diseñadores de compiladores se preocupan más por proporcionar mejores posibilidades, que a un lenguaje estándar, sin embargo, las mejoras no son estándar para los demás.

Otro aspecto es la disponibilidad de herramientas de desarrollo puede acortar el tiempo que se requiere para la generación de código fuente y puede mejorar la calidad del código. Muchos de los lenguajes de programación pueden ser adquiridos con un conjunto de herramientas las cuales ayudan de manera significativa al desarrollador y hacen que la generación del código sea mucho más rápida y eficiente.

Por último la facilidad de mantenimiento del código fuente es muy importante, debido a que el mantenimiento no se puede llevar a cabo hasta que se entiende el software. Para ello es necesario que el software en cuestión cuente con documentación de diseño para proporcionar facilidad en la comprensión del mismo, ya que el código fuente final debe ser leído y modificado de acuerdo con los cambios en el diseño. La facilidad de traducción del diseño al código fuente es un elemento importante en la facilidad de mantenimiento del mismo. Además, las características mismas de la documentación de un lenguaje (por ejemplo, la longitud de identificadores no limitada, etiquetado, definición de estructuras y de tipos de datos), tienen una fuerte influencia sobre el mantenimiento.

La elección de un lenguaje de programación para un proyecto específico debe tener en cuenta principalmente las características de ingeniería, sin embargo, el problema asociado con la elección de un lenguaje puede desaparecer si sólo se dispone de un lenguaje o si el cliente demanda uno en particular.

Independientemente del paradigma de ingeniería del software, el lenguaje de programación tendrá impacto en la planificación, el análisis, el diseño, la codificación, la prueba y el mantenimiento de un proyecto. Los lenguajes de programación proporcionan los medios para la traducción hombre-máquina; sin embargo, la calidad del resultado final se encuentra unida a las actividades de la ingeniería de software que preceden y siguen a la codificación.

Las características técnicas de los lenguajes de programación abarcan un gran número de aspectos que van desde lo teórico hasta lo práctico. Los fundamentos de los lenguajes de programación los enfocaremos principalmente a cuatro grandes áreas: tipificación de datos, mecanismos de subprograma, estructuras de control y soporte para métodos orientados a objetos. Cabe señalar que todos los lenguajes de programación se pueden caracterizar de acuerdo con estos aspectos y se puede juzgar la calidad global de un lenguaje de programación específico evaluando la potencia o debilidad de cada uno de ellos.

### **2.1.1 Tipos de datos y tipificación de datos**

La tipificación de datos y los tipos de datos específicos que son soportados por un lenguaje de programación son un aspecto importante que influyen en la calidad del lenguaje. Los tipos de datos simples abarcan un amplio rango que incluye los tipos numéricos, tipos enumerados, tipos lógicos y tipos de cadena. Los tipos de datos más complejos son aquellos que engloban estructuras de datos que van desde los arrays (vectores) simples unidimensionales hasta las estructuras de listas y los complejos heterogéneos arrays y registros.

Las operaciones que se pueden realizar sobre un tipo de datos particular y la forma en que se pueden manipular distintos tipos en una misma sentencia se controlan por la comprobación de tipos incorporada en el intérprete o compilador del lenguaje de programación.

## **2.1.2 Subprogramas**

Los subprogramas son componentes que forman parte de un programa, estos componentes son compilados por separado, además de que contienen datos y estructuras de control. Sin considerar el nombre del subprograma, este cuenta con ciertas características: una sección de especificación que incluye su nombre y la descripción de la interfaz; una sección de implantación que incluye los datos y las estructuras de control y por último un mecanismo de activación que permite que el subprograma sea requerido desde cualquier otro punto del programa.

En los lenguajes de programación orientados a objetos, el punto de vista clásico del subprograma es reemplazado por el de objeto.

## **2.1.3 Estructuras de control**

En un nivel básico y fundamental, todos los lenguajes de programación modernos permiten al programador representar secuencias, condiciones y repeticiones, a lo cual llamamos propiamente como estructuras de control. Existen otras estructuras como la recursividad, la cual crea una segunda activación de un subprograma durante la primera activación. Por otro lado la concurrencia da soporte a la creación de múltiples tareas, a la sincronización de las mismas y a la comunicación de las tareas en general. Por último tenemos el manejo de excepciones que es una característica del lenguaje la cual atrapa los errores del sistema o los definidos por el usuario, pasando el control a un manipulador de excepciones para que los procese.

## **2.1.4 Soporte para el enfoque orientado a objetos**

En teoría, la creación de los objetos y la construcción de software orientado a objetos se puede llevar a cabo utilizando cualquier lenguaje de programación convencional. Aunque en la práctica, el soporte para métodos orientados a objetos debe estar

incorporado directamente en el lenguaje de programación que se vaya a utilizar para implementar un diseño orientado a objetos.

Otras de las características que la programación orientada a objetos debe proporcionar es un soporte directo para la definición de clases, la herencia, la encapsulación y el paso de mensajes. Además de estas construcciones básicas, muchos de estos lenguajes implantan características adicionales, como herencia múltiple y polimorfismo.

## 2.2 Bases de Datos.

En ésta sección nos ocuparemos de definir los conceptos básicos de Bases de Datos Relacionales, dada la importancia del modelo relacional (introducido por Codd en 1970<sup>1</sup>) en el desarrollo del presente trabajo; pues los sistemas relacionales ofrecen muchos tipos de procesos de datos, tales como: simplicidad y generalidad, facilidad de manejo para el usuario final, periodos cortos de aprendizaje y las consultas de información se especifican de forma sencilla.

Comenzaremos definiendo un modelo de datos como un sistema formal y abstracto, que permite describir los datos de acuerdo con unas reglas y convenios predefinidos. Es formal en el sentido de que los objetos del sistema se manipulan siguiendo unas reglas perfectamente definidas, y utilizando exclusivamente los operadores definidos en el sistema, independientemente de lo que estos objetos y operadores puedan significar<sup>2</sup>. Cabe señalar, que un modelo de datos cuenta con tres componentes:

- Estructura de datos: Se refiere a una colección de objetos abstractos formados por datos.
- Operadores entre las estructuras: Conjunto de operadores, con reglas bien definidas, que permiten manipular las estructuras de datos.
- Definiciones de integridad: Colección de conceptos y reglas que permite expresar qué valores de datos pueden aparecer válidamente en nuestro esquema.

Hay varios modelos de datos, entre ellos los más utilizados son: jerárquico, red, relacional y orientados a objetos. Y como se dijo en un principio, aquí nos ocuparemos del estudio del modelo relacional.

<sup>1</sup> Sistemas de Bases de Datos, pág. 140.

<sup>2</sup> Bases de Datos Relacionales, pág. 37.

## 2.2.1 Estructura de datos relacional.

### Dominio

Un dominio  $D$  es un conjunto finito de valores homogéneos y atómicos,  $V_1, V_2, \dots, V_n$  caracterizados por un nombre; decimos valores homogéneos porque son todos del mismo tipo, y atómicos porque son indivisibles en lo que al modelo se refiere<sup>3</sup>.

### Atributos

A cada dominio que participa en una relación se le da un nombre que lo identifica de forma única para esa relación: en términos informales, los atributos de una relación corresponden a las columnas de la tabla<sup>4</sup>.

### Relación

Matemáticamente, una relación definida sobre los  $n$  dominios  $D_1, D_2, \dots, D_n$ , es un subconjunto del producto cartesiano de estos dominios, donde cada elemento de la relación, tupla, es una serie de  $n$  valores ordenados<sup>5</sup>.

De tal forma, si colocamos todas las tuplas de una relación una debajo de otra, alineando los componentes correspondientes de cada atributo en una columna, y colocamos en la cabecera de cada columna el nombre de cada atributo, obtenemos una representación de la relación: una tabla de dos dimensiones<sup>6</sup>. Donde el número de filas de una relación se denomina cardinalidad, mientras que el número de columnas es el grado.

---

<sup>3</sup> Concepción y Diseño de Bases de Datos, pág. 432

<sup>4</sup> Bases de Datos Relacionales, pág. 20

<sup>5</sup> Concepción y Diseño de Bases de Datos, pág. 436

<sup>6</sup> Idem

Como se mencionó anteriormente una relación se puede representar en forma de tabla, aunque tiene una serie de elementos característicos que la distinguen de la tabla:

- No puede haber filas duplicadas, es decir, todas la tuplas tienen que ser distintas.
- El orden de las tuplas (filas) no es significativo.
- El orden de los atributos (columnas) no es significativo.
- La tabla es plana, es decir, en el cruce de una fila y una columna sólo puede haber un valor (no se admiten atributos multivaluados).

### 2.2.2 Manipulación de Datos.

Enseguida analizaremos el álgebra relacional, cuya función es la de servir como modelo teórico de la capacidad de recuperación de información que todo lenguaje de consulta debe poseer como mínimo<sup>7</sup>.

#### Álgebra Relacional

El álgebra relacional consiste de un conjunto de operadores que se aplican a una o dos relaciones y producen como resultado otra operación. Vamos a describir a continuación estos operadores.

Unión:  $R \cup S$ .

El resultado es una relación que incluye a todas las tuplas de  $R$  y todas de  $S$ , y ninguna más. Si hubiera alguna repetida en  $R$  y  $S$ , sólo figurará una vez en el resultado. Nótese, que  $R$  y  $S$  deben ser relaciones del mismo grado<sup>8</sup>.

---

<sup>7</sup> Introducción a las Bases de Datos Relacionales, pág. 29.



Diferencia :  $R - S$ .

Cuando es aplicado este operador el resultado es una relación que incluye a todas las tuplas de  $R$  que no están en  $S$ . En este caso también  $R$  y  $S$  deben ser del mismo grado<sup>9</sup>.

Intersección:  $R \cap S$ .

Cuyo resultado es una relación que incluye a todas las tuplas de  $R$  que también están en  $S$ . De igual forma que en los casos anteriores,  $R$  y  $S$  deben ser relaciones del mismo grado<sup>10</sup>.

Producto:  $R \times S$ .

En esta operación, conocida también como PRODUCTO CARTESIANO, el resultado incluye a todas las tuplas posibles que se obtienen concatenando una de  $R$  con otra de  $S$ <sup>11</sup>. Respecto a la cardinalidad de  $R \times S$ , esta es  $ab$  si  $a$  es la cardinalidad de  $S$  y  $b$  de  $R$ . El grado de  $R \times S$  se define como  $x+y$  si  $x$  e  $y$  son grados de  $R$  y  $S$  respectivamente<sup>12</sup>.

Cociente:  $R / S$ .

El concepto de este operador esta relacionado con el producto cartesiano en el que  $R \times S$  dividido por  $S$  da como resultado  $R$ . Dividir sin embargo, es esencialmente el inverso del producto cartesiano.

Supongamos que  $R$  es una relación binaria y  $S$  es unaria, el dominio del segundo atributo de  $R$  es el mismo del de  $S$ .  $R/S$  es entonces la relación con un solo atributo

---

<sup>8</sup> Bases de Datos Relacionales, pág. 24

<sup>9</sup> Idem

<sup>10</sup> Idem

<sup>11</sup> Idem

que tiene el mismo dominio del primer atributo de R y cada tupla en R/S ocurre en R concatenando con cada tupla de S<sup>13</sup>.

Selección. S(F) (R): Selección de la relación de acuerdo con la fórmula F.

Formalmente, la selección de una relación R es un subconjunto de la relación S de las tuplas en R, donde esas tuplas extraídas cumplen cierta condición expresada por la fórmula F. Dicha fórmula se construye con operandos que pueden ser constantes, o atributos de R, ligados entre si por los operadores definidos sobre los respectivos dominios. El grado de S es el mismo que el de R. Y la cardinalidad de S es igual al número de tuplas en R que satisfagan una condición específica<sup>14</sup>.

Proyección:  $\pi$  s[R].

La operación de proyección es la selección de ciertos atributos de una relación R para formar una nueva relación de S, para ello se eliminan aquellas tuplas que se encuentran duplicadas. La proyección de R en los atributos a,b,c, está denotada por R[a, b, c], algunos autores utilizan  $\pi$  a , b, c[R] para denotar la proyección, para generalizar esta notación tenemos  $\pi$  s[R] donde R es la relación y s la lista de atributos a proyectar<sup>15</sup>.

Yunción: Se representa R Y S(i Cj).

Se lee: Yunción de R y S, mediante el operador de comparación C sobre las columnas de i de R y j de S. Donde, C es un operador de comparación (>, <, =, >=, <=,  $\neg$  =). E i y j son atributos de R y S, respectivamente<sup>16</sup>.

---

<sup>12</sup> Tesis profesional de Ingeniería en Computación, UNAM, pág. 31

<sup>13</sup> Idem, pág. 32

<sup>14</sup> Bases de Datos Relacionales, pág. 26.

<sup>15</sup> Tesis profesional de Ingeniería en Computación, UNAM, pág. 33

<sup>16</sup> Idem, pág. 27

El resultado es el conjunto de todas la tuplas formadas concatenando una de R con otra de S de manera que se cumpla la condición (i C j).

*Equiunción:* Cuando la comparación es por igualdad (o sea, el operado C es el signo = ).

*Yunción natural:* La representaremos por ( R Y S )

Sean dos relaciones R y S que tienen algunos atributos iguales (es decir, columnas homónimas). La yunción natural es el resultado de :

- Concatenar todas las tuplas de R y S (en otras palabras, el producto  $R \times S$ ).
- Seleccionar de entre estas tuplas concatenadas las que tengan iguales valores en todas sus columnas homónimas.
- Suprimir una columna de cada dos homónimas en el resultado.

Si R y S no tienen atributos comunes, la yunción natural será igual al producto:

$$R * S = R \times S.$$

### 2.2.3 Reglas de Integridad.

En este punto estudiaremos brevemente los diversos tipos de restricciones que se pueden especificar en un esquema de bases de datos relacional. Entre estas restricciones se encuentran la de clave, de integridad de entidades y de integridad referencial.

#### Restricciones de clave.

Recordemos que en una relación R todas sus tuplas han de ser diferentes, es decir, no se repiten sus valores. De tal forma, que a todo conjunto de atributos con esa

propiedad se le llama superclave<sup>17</sup>. Sin embargo, una superclave puede tener atributos redundantes, así que un concepto más útil es el de clave, que carece de redundancia. Una clave K de un esquema de relación R es una superclave de R con la propiedad adicional que la eliminación de cualquier atributo A de K deja un conjunto de atributos K que no es una superclave de R<sup>18</sup>, pues incumpliría con la restricción de unicidad.

Entre los valores que toma una clave de una relación y las tuplas de ésta existe una correspondencia biunívoca, de modo que las claves pueden utilizarse como identificadores de las tuplas de la relación<sup>19</sup>. En el modelo relacional, la única forma de encontrar una tupla determinada en una relación, es conociendo el valor de una clave.

En general, un esquema de relación puede tener más de una clave. En tal caso, cada una de ellas se denomina:

- Clave candidata. Es aquella super clave que no contiene atributos ajenos, es decir, aquel conjunto de atributos que no tiene un subconjunto menor que pueda considerarse como super clave<sup>20</sup>.
- Clave primaria. Es aquella llave que el diseñador de la base de datos selecciona entre las llaves candidatos encontradas<sup>21</sup>.

### Restricción de integridad de entidades

Establece que ningún valor de clave primaria puede ser nulo. Esto es porque el valor de la clave primaria sirve para identificar las tuplas individuales en una relación; el que la clave primaria tenga valores nulos implica que no podemos identificar algunas tuplas<sup>22</sup>.

### Restricción de integridad referencial.

<sup>17</sup> Bases de Datos Relacionales, pág. 20.

<sup>18</sup> Sistemas de Bases de Datos, pág. 146

<sup>19</sup> Bases de Datos Relacionales, pág. 21.

<sup>20</sup> Tesis profesional de Ingeniería en Computación, UNAM, pág. 36.

<sup>21</sup> Idem.

<sup>22</sup> Sistemas de Bases de Datos, pág. 149.

La restricción de integridad referencial se especifica entre dos relaciones y sirve para mantener la consistencia entre tuplas de las dos relaciones<sup>23</sup>.

### Formas normales

Las formas normales son técnicas cuyo objetivo es eliminar los datos redundantes para evitar anomalías en las tablas. Para la normalización de datos se establecen reglas, las cuales se conocen como primera, segunda y tercera formas normales<sup>24</sup>.

- La Primera Forma Normal (FNF) establece que un atributo no puede contener múltiples valores, es decir, que los únicos valores de atributos que permite FNF son valores atómicos (o indivisibles).
- La Segunda Forma Normal (SNF) establece que toda columna que no sea clave debe depender por completo de la clave primaria, y no sólo de parte de ésta. Para que una tabla esté en la Segunda Forma Normal, también debe estar en conformidad con la Primera Forma Normal.
- La Tercera Forma Normal (TNF), al igual que la SNF, establece que todas las columnas que no sean clave no deben depender de ninguna otra que no sea clave. No se debe olvidar que la tabla también debe estar en conformidad con la Segunda Forma Normal.

### Razón de cardinalidad.

Un concepto del modelo Entidad – Relación muy útil en el modelo Relacional es la razón de cardinalidad, que es aquella mediante la cual puede especificarse la cantidad de entidades que podrán asociarse mediante una relación. La razón de cardinalidad se aplica generalmente sobre dos conjuntos de entidades<sup>25</sup>.

---

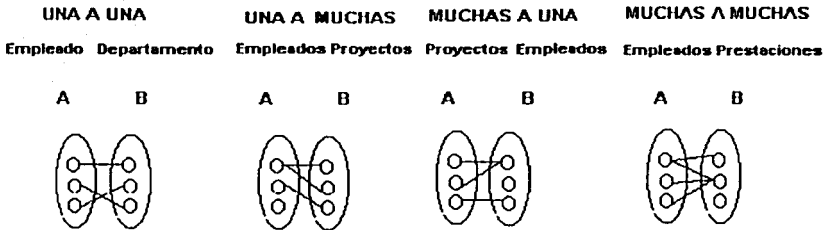
<sup>23</sup> Idem.

<sup>24</sup> Aprendiendo Microsoft SQL Server 2000, pág. 17

<sup>25</sup> Tesis profesional de Ingeniería en Computación, UNAM, pág. 35

Las cardinalidades existentes para dos conjuntos de entidades A y B y conjunto de relaciones R pueden ser:

1. **UNA A UNA.** Una entidad de A puede asociarse únicamente con una entidad B.
2. **UNA A MUCHAS:** Una entidad de A puede asociarse con cualquier cantidad de entidades de B.
3. **MUCHAS A UNA:** Cualquier cantidad de entidades de A puede asociarse con un entidad de B.
4. **MUCHAS A MUCHAS:** Cualquier cantidad de A puede asociarse con cualquier cantidad de entidades de B.



**Figura 2.2.3.1 Tipos de Relaciones**

## **2.3 Metodología de Base de Datos y Lenguajes de Programación Orientada a Objetos.**

### **2.3.1. Metodología de Bases de Datos.**

Una metodología es un conjunto de modelos y herramientas que nos obliga a trabajar de forma organizada. Existen diversas metodologías para un diseño de base de datos tales como: MERISE, SSADM y Yourdon, entre otras.

#### **MERISE**

Es un método de enfoque sistemático, en el cual se consideran 4 etapas:

- **Estudio Preliminar:** Aquí es donde se recopilan los datos, se conceptualiza la solución en forma global; se evalúa y se establece un borrador del plan de desarrollo.
- **Estudio detallado:** Permite definir explícitamente las especificaciones funcionales de la aplicación a través de una conceptualización en forma general de sistema, posteriormente la conceptualización detallada de las fases y un refinamiento en el plan de desarrollo.
- **Realización:** Esta etapa consiste en el estudio técnico a través de la distribución de datos en archivos físicos y el tratamiento de programas y de Producción.
- **Puesta en Marcha:** Consiste en la Preparación de los recursos; y Recepción y lanzamiento del sistema. Es decir, la formación del personal, lanzamiento de la aplicación y recepción definitiva por parte del Usuario.
- **Mantenimiento:** Esta etapa abarca el resto de la vida útil de la Base de Datos.

## SSADM

Esta metodología proporciona un conjunto de especificaciones y procedimientos para llevar a cabo las tareas de análisis y diseño de la base de datos. Se reconocen 3 Fases en etapas:

- Estudio de Viabilidad, el cual se desarrolla en 2 etapas: Definición de Problema e Identificación del Proyecto
- Análisis: Análisis del sistema actual, Especificación de requerimientos, Selección de Opciones técnicas.
- Diseño: Diseño de Datos, Diseño de procesos y Diseño físico.

## Yourdon.

La metodología se basa en un análisis de diseño estructural; utiliza el método descendente(**top-down**) y descomposición funcional para definir los requerimientos del sistema y diseño de bases de datos. El enfoque estructurado permite modelar las componentes del sistema por medio de símbolos gráficos, estableciendo una atención especial en los aspectos lógicos del sistema. La descripción de los requerimientos se convierte en la liga entre el análisis y el diseño. Yourdon establece 3 Niveles para el desarrollo de un sistema: Nivel Conceptual que comprende las Especificaciones; El Nivel Lógico para el Análisis Lógico; y el Nivel Físico que contiene el Diseño Físico, la Implantación y el Mantenimiento; los cuales se detallaran mas adelante.

La metodología de Yourdon esta basada principalmente en los siguientes conceptos:

- Diagramas de flujo de datos para la representación de procesos.
- Diagramas de transición de estados para la representación estructurada de las funciones a realizar en los procesos.
- Modelo Entidad/Relación para la representación conceptual de datos.
- Diccionario de datos como base o soporte de información del sistema.



- Diagramas o mapas de estructura para la representación modular de los procesos y las variables intercambiadas entre ellos.
- Especificaciones de programas basadas en lenguaje estructurado y tablas de decisión.
- Diagrama Entidad-Relación (DER).
- Normalización.

La metodología de Yourdon consiste en un proceso sistematizado y a la vez flexible en cuanto a la aplicación de las técnicas estructuradas, Esta metodología se definen en 3 etapas: Modelado de Procesos, Modelado de Datos y Etapas del ciclo de vida.

### **Modelado de procesos**

La etapa de análisis y especificaciones estructurada, examina los problemas mediante el modelo entrada-proceso-salida, es decir mediante el flujo de información. En esta etapa son utilizados: el diagrama de flujo de datos, el diccionario de datos, diagramas de transición de estados, las especificaciones y las tablas de decisión.

El Diagrama de Flujo de Datos (DFD), permite crear una descripción gráfica del problema, que establece los caminos que seguirán los datos a través de los procesos del sistema. Para elaborar un DFD eficaz tenemos que seguir los siguientes pasos:

- Desarrollar el DFD mediante el enfoque descendente: Hacer una lista de las entidades, los flujos de datos y los procesos y almacén de datos del sistema. Dibujar un DFD básico que ilustre exclusivamente los aspectos generales. Este diagrama deberá incluir lo básico de las entradas, salidas y procesos del sistema.
- Cubrir detalles: Añadir, por pasos, los detalles de cada proceso. Indicar excepciones, siempre que sea necesario. Aquí el diagrama original debe descomponerse en diagramas que detallan los procesos, agregando en cada nivel inferior nuevos almacenes de datos y flujos de datos.

- Dibujar de nuevo los diagramas y volver a definir todos los símbolos por medio de nombres significativos.

Diccionario de Datos, es una referencia útil para localizar nombres y atributos de los datos utilizados en el sistema. Los registros del diccionario de datos deben contener información referente al dato tales como: Nombre y Alias; descripción del contenido del dato, flujos de datos que se relacionan con el término; longitud del dato y cualquier otra información pertinente.

Diagramas de transición de estados, nos permiten representar diferentes funciones a realizar indicando la secuencia y las condiciones que manejan. Cada proceso debe verse representado por una serie de acciones enmarcadas en rectángulos. Se pasa de una acción a la siguiente a través de una conexión en la que se representa el par condición- acción.

Especificación estructurada, sugiere que el esquema lógico de cualquier programa debería ser escrito con un conjunto limitado de estructuras de control, que se componen de verbos imperativos y palabras reservadas para expresar la lógica del funcionamiento.

Tablas de decisión, empleadas para traducir las acciones y condiciones a una forma tabular.

Diagramas o Mapas de Estructura: Representan gráficamente la estructura de un proceso descomponiéndolo en módulos que se intercambian y se encadenan hasta completar las funciones a realizar.

Diagrama Entidad-Relación: Representa gráficamente las relaciones entre las diferentes entidades, como también sus atributos y las características de éstos

Normalización: Consiste de una serie de pasos ordenados, conocidos como Formas Normales(FN); cuyo fin es simplificar las relaciones existentes entre los datos de las

aplicaciones y los almacenamientos usados, de tal manera que se establezcan estructuras de menos tamaño. Muy brevemente, en la 1FN<sup>26</sup> tiene como objetivo eliminar los grupos de datos repetidos e identificar una llave primaria; en la 2FN<sup>27</sup> se debe eliminar cualquier dependencia de los atributos con la llave primaria y 3FN<sup>28</sup> se debe eliminar cualquier dependencia transitoria.

### **Modelado de datos**

Esta técnica representa las necesidades del usuario para un sistema en función de los datos del sistema independientemente de cómo se procesen o utilicen los datos.

Se basa principalmente en la Base de Datos, entonces es necesario considerar el modelo Entidad-Relación, Yourdon propone un esquema de entidades relacionales del sistema, pasando después a normalizar la estructura. En esta etapa se aplican las tres técnicas de primera forma normal (1FN), segunda forma normal(2FN) y tercera forma normal(3FN).

### **Ciclo de Vida**

- **Especificaciones:** Consiste en la descripción de la situación actual del problema, es decir, definir las necesidades y objetivos del sistema.
- **Análisis Lógico:** Se utilizan todas las técnicas de diagramación y modelado de procesos.
- **Diseño Físico:** Consiste en la representación lógica de los datos mediante un diseño dependiente del modelo de base elegida una estructuración de los procesos utilizando diagramas de estructura de los mismos y generando las especificaciones de los programas.
- **Implantación:** Comprende la generación del código y el ensamblaje e integración a todos los módulos.

---

<sup>26</sup> 1FN Primera Forma Normal

<sup>27</sup> 2FN Segunda Forma Normal

<sup>28</sup> 3FN Tercera Forma Normal

- **Mantenimiento.** Se refiere a los cambios realizados al sistema para incluir una mejora o una corrección al código.

### 2.3.2. Metodología Orientada a Objetos.

Los métodos de análisis y diseño Orientados a Objetos se pueden desglosar en dos tipos básicos: Los que reproducen métodos estructurados ya existentes utilizando tres notaciones distintas para datos, dinámica y procesos; y aquellos otros que afirman que, dado que los objetos combinan de modo innato los procesos (métodos) y los datos, solamente es necesaria una notación. Comúnmente estas dos aproximaciones se llaman la TERNARIA (o trilateral), la cual tiene una ventaja: ya que posee notaciones tradicionales, la transición hacia su uso es más suave; y la UNARIA, que por tener notación abstracta, es más fácil aprender desde cero.

De las aproximaciones o métodos trilaterales, las más conocidas y claras son la OMT, la Ptech modificada de Martín y Odell y la OSA. De las unarias las más conocidas son la CRC, la Coad/Yourdon y la más reciente de ellas la UML( Lenguaje Unificado de Construcción de Modelos). Además podemos mencionar otros métodos híbridos de ambos tipos como Herderson-Seller y SOMA.

Todas la metodología ofrecen ciertas ventajas y desventajas; por ejemplo: el método más popular de Coad/Yourdon, siendo el más sencillo, carece de apoyo para describir la dinámica de sistemas; La OMT es muy difícil de aprender, pero no permite expresar claramente reglas o limitaciones propias del negocio; OSA y la síntesis IE con Ptech, son sencillas de aprender y permiten al analista escribir limitaciones en los diagramas, como si fuera un pensamiento posterior, pero no admiten reglas. Una de las deficiencias de todos los métodos existentes de análisis Orientado a Objetos es la forma en que se añadieron las reglas y limitaciones, así como el pensamiento posterior.

A continuación haremos una breve revisión a los métodos mencionados anteriormente:

---

## OOSA de Shlaer/Mellor

Fue uno de los primeros ejemplos del análisis orientado a objetos, pero en su primera definición(1988) no podría ser considerado como Método Orientado a Objetos, pues ésta carecía de ideas de herencia. Fue hasta su segunda definición donde se incluyeron los principios de herencia (subtipos de entidades). Este método puede clasificarse como trilateral, y se desarrolló creando un modelo de información (o datos) que muestra objetos, atributos y relaciones. A continuación un modelo de estado describe los estados de los objetos y las transiciones entre estados. Por último, un DFD define el modelo de procesos. El primer paso de este método es la definición de objetos y sus atributos. La notación de modelado de entidades ésta basada en la definición usada por Ward/Mellor. Se hace hincapié en la definición de los historiales vitales de los objetos utilizando unos diagramas de transición de estados y las tablas correspondientes. La dinámica global se administra coordinando los ciclos vitales de los objetos anotados en modelos de comunicaciones, que muestran sucesos, -mensajes- que pasan entre entidades. A las relaciones también se les da el nombre de modelos de ciclo vital.

## Coad/Yourdon

Surge en 1990. Tiene un método de análisis y una notación de apoyo razonablemente completos, prácticos y orientados a objetos, adecuados para proyectos comerciales. El análisis mediante esta metodología se produce en 5 fases:

- **Temas:** Descomponer el problema en "TEMAS", que corresponden a niveles o capas en los diagramas de flujo. Se recomienda que estos temas sean de tamaño manejable, entre 5 a 9 objetos.
- **Objetos:** Identificar objetos con detalle, buscando entidades de negocio.
- **Estructuras:** Se identifican dos estructuras: las de clasificación, que se relacionan directamente con la herencia y las de composición,
- **Atributos:** Son detallados y se especifican las relaciones de modalidad y de multiplicidad empleando una versión del análisis relacional extendido (ERA)

- **Servicios:** Que son las operaciones realizadas, Cada tipo de Objeto debe estar equipado con un Método para crear y borrar instancias, para tomar y poner valores y, también deberá disponer de métodos especiales que caractericen su comportamiento.

Coad y Yourdon ofrecen el código mnemotécnico "SOSAS" para ayudar a recordar estos pasos.

## OMT

La técnica de Modelado de Objetos(OMT), es considerada como una de las más complejas, se trata de una aproximación trilateral con fuertes raíces en los métodos estructurados tradicionales, y ofrece una notación muy rica, por lo cual también muy complicada. Por lo tanto resulta recomendable el apoyo automatizado, que puede llegar a generar el código, como lo hacen algunas herramientas; este método esta normalmente asociado a desarrollos en C++ y Smalltalk

El análisis mediante esta metodología se produce en 3 fases:

- **Análisis:** Primero, se construye el Modelo de Objetos (OM) que consta de diagramas y de un diccionario de datos. A continuación se construye el Modelo Dinámico(DM) que consta de STD y de diagramas de flujo globales; y por último es la construcción del modelo funcional
- **Diseño de Sistemas:** Se realiza organizando los objetos en subsistemas, identificando la concurrencia a partir del Modelo Dinámico
- **Diseño de Objetos:** Implica transformar la información del Modelo Dinámico y del Modelo Funcional a un Modelo de Objetos.

## UML Unified Modeling Language<sup>29</sup>

Esta metodología está enfocada a la construcción de Modelos Orientados a Objetos, que se apoya en su mayor parte en una notación esquemática. Esta metodología tiene un conjunto de patrones utilizados en distintas fórmulas de solución de problemas y utiliza las mismas fases de tradicionales de desarrollo de una aplicación, que son: Planeación y Elaboración, Construcción y Aplicación. El punto focal de esta metodología es el uso de un **ciclo de vida iterativo**, que se basa en el agrandamiento y perfeccionamiento secuencial de un sistema a través de múltiples ciclos de desarrollo, análisis, diseño, implantación y pruebas.

El sistema crece al incorporar nuevas funciones en cada ciclo de desarrollo. Tras una fase preliminar de planeación y especificación, el desarrollo pasa a una fase de construcción a través de una serie de ciclos de desarrollo. En cada ciclo se aborda un conjunto relativamente pequeño de requerimientos para cada fase de desarrollo.

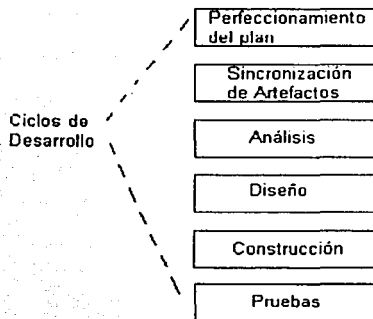


Figura 2.3.2.1 Ciclo de desarrollo.

Cada ciclo de desarrollo está compuesto de múltiples tareas o etapas como se muestra en la figura 2.3.2.1. Y a su vez, cada etapa o tarea contiene un conjunto de pasos

<sup>29</sup> UML Lenguaje Unificado de Construcción de Modelos

específicos para completarse. La figura 2.3.2.2 muestra algunas de estas tareas para las etapas de Análisis y Diseño.

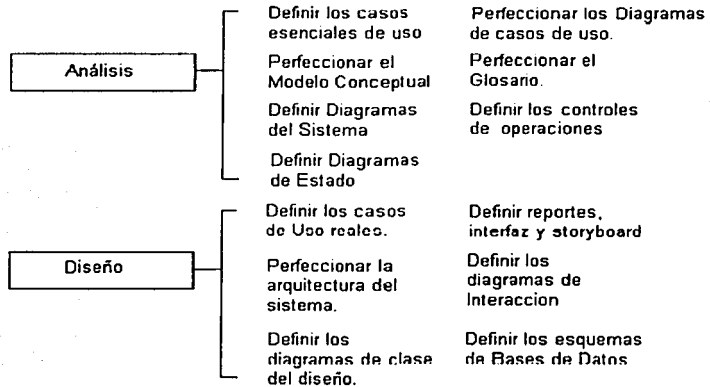


Figura 2.3.2.2 Tareas a desarrollar dentro de un ciclo.

Se describirá brevemente cada uno de estos conceptos:

- **Perfeccionamiento del Plan:** Se debe de establecer un programa, en el cual se asignen los recursos, las prioridades y el presupuesto; También, se establece la declaración inicial de los requerimientos y un glosario inicial de nombres y conceptos y toda la información afin.
- **Sincronización de Artefactos:** Corresponde a la descomposición de los modelos en submodelos que contienen funciones mas concretas del ciclo de desarrollo que se está estudiando. Dentro de la metodología UML, los artefactos son los diagramas de casos de uso y los diagramas de interacción.
- **Etapas de Análisis:** Se deben identificar y describir los objetos o conceptos dentro del dominio del problema. Dentro de análisis se debe desarrollar diversas tareas como: Identificar los procesos y las necesidades del negocio expresándose en Casos de



Uso (los cuales son descripciones narrativas textuales de los procesos del sistema); Expresar las diversas categorías de cosas del dominio a través de un modelo conceptual; Distribuir las funciones y las responsabilidades de varios objetos de software en el sistema, que se expresan a través de diagramas de diseño de clase y con diagramas de colaboración en los cuales se muestra la definición de las clases y el flujo de mensajes entre los objetos de software.

- Etapa de Diseño: Se deben definir los objetos lógicos del sistema que finalmente serán implantados en el Lenguaje de Programación Orientada a Objetos.
- Construcción: Se debe implantar los componentes de diseño como una clase.

## 2.4 Características, ventajas y desventajas de SQL Server 2000

Microsoft SQL Server (Servidor) es un sistema administrador de base de datos Relacional, basado en la arquitectura Cliente/Servidor, además de ser un motor de bases de datos, que funciona en el sistema operativo NT. Su lenguaje estructurado de consulta SQL (Structured Query Language), es un estándar informático corrientemente utilizado para definir, modificar y gestionar datos, además de controlar los cambios en la base de datos usando tablas, índices, claves, que permiten almacenar la información. SQL fue desarrollado a partir de las ideas del Dr. Edgar F. Codd, de IBM, que colaboró en el desarrollo del modelo relacional cuando trabajaba en los laboratorios de investigación de IBM en los años 70.

SQL Server utiliza la arquitectura Cliente/Servidor para distribuir la carga de trabajo en tareas que corran en computadoras tipo Servidor y tareas que corran en computadoras tipo cliente.

El cliente es el responsable de la parte lógica y de presentar la información al usuario. Generalmente, el cliente corre en una o más computadoras Cliente, aunque también puede correr en una computadora Servidor con SQL Server.

Una aplicación *cliente/servidor* puede definirse como aquella que se divide en dos partes: una parte se ejecuta en un servidor y la otra se ejecuta en el cliente o estaciones de trabajo. El lado del servidor de la aplicación proporciona la seguridad, la tolerancia a fallas, el desempeño, la concurrencia y las copias de seguridad confiables. El lado del cliente proporciona la interfaz de usuario y puede contener informes, aprovechando los dos y juntándolos.

SQL Server es la parte del servidor de la ecuación: hay varios clientes de donde elegir para conectarse a SQL Server, incluyendo las utilerías que vienen con éste, como el Analizador de Consultas de SQL Server (SQL Server Query Analyzer).

SQL Server proporciona las siguientes ventajas tanto para clientes como para servidores.

**Ventajas del cliente:**

- Facilidad de uso
- Maneja múltiples plataformas de hardware
- Maneja múltiples aplicaciones de software
- Familiar al usuario

**Ventajas del servidor:**

- Confiable
- Concurrente
- Bloqueo sofisticado
- Tolerante a fallas
- Hardware de alto desempeño
- Control centralizado

En la computación cliente/servidor, cuando se ejecuta una consulta, el servidor examina la base de datos y envía al cliente sólo las filas que corresponden. Esto no sólo ahorra tráfico en el ancho de banda de la red, sino que puede ser más rápido que hacer que las estaciones de trabajo realicen la consulta, siempre que el servidor sea una máquina lo suficientemente poderosa.

SQL Server administra bases de datos de tipo relacional y distribuye los recursos disponibles del servidor (tales como memoria, operaciones de disco, entre otros) entre las múltiples peticiones.

Otra característica de SQL Server es que, ahora, una única base de datos puede residir en múltiples archivos por medio de la característica de grupos de archivos, aunque no podrán coexistir múltiples bases de datos en un solo archivo. Esto conlleva una gran mejora en el rendimiento y la capacidad.

Microsoft ha tomado el núcleo tecnológico de la base de datos SQL Server y lo ha rediseñado con mayor capacidad de almacenamiento de datos, un mejor optimizador y una nueva versión del motor de almacenamiento de la base de datos. También es importante la mejora en cuanto a su capacidad de crecimiento, un objetivo continuo de SQL Server. SQL Server puede ahora almacenar un terabyte en cada base de datos. El concepto empresarial de Microsoft es intentar conocer las necesidades del mercado de bases de datos de gama alta en el entorno de la computación distribuida y suministrar todas las herramientas necesarias para manejar los muchos servidores exigidos en este tipo de entorno, incluyendo la capacidad para acceder ocho procesadores. Todo esto se agrupa en la MMC (Microsoft Management Console). La funcionalidad de apropiación (snap in) de la MMC permite la inserción de herramientas de otros fabricantes para manejar el entorno y se suministra con toda funcionalidad de Enterprise Manager, el cual está incluido en el paquete.

SQL Server revoluciona el concepto de base de datos, reúne en tan sólo un producto la potencia necesaria para cualquier aplicación de tipo empresarial junto con unas herramientas de gestión, las cuales reducen al mínimo el costo de la propiedad. El uso de este paquete de Microsoft permite a la empresa tener todo en serie. También se posee la libertad de elección, debido a que todas las aplicaciones de gestión del mercado corren en este manejador de base de datos.

SQL Server se adapta a las necesidades de la empresa, soportando desde unos pocos usuarios a varios miles, también podemos señalar que es la mejor base de datos para trabajar bajo el ambiente de Windows NT. Posee los mejores registros de los benchmarks (evaluación de laboratorio) independientes tanto en transacciones totales

---

como el costo por transacción. SQL Server también es compatible con Visual Basic, Visual C++, Visual J++, Visual Interdev, Microfocobol y otras herramientas más. SQL Server permite llamadas a procedimientos remotos Servidor/Servidor, una relación asíncrona o continua basada en registros, o sincronización planificada de tablas. Permite la configuración de replica en forma gráfica y característica de gestión, replica a suscriptores ODBC, incluyendo IBM DB2, ORACLE, SYBASE y Microsoft Access. El Distributed Transaction Coordinator (Coordinador de Transacción Distribuida) interviene en transacciones que involucran dos o más servidores SQL.

En cuanto a la gestión y administración centralizada de bases de datos SQL Enterprise Manager, es una consola de gestión y motorización de 32 bits visual basada en Windows, posee un único punto de configuración y gestión de control de datos remotos. También con el SQL Executive (Ejecutivo), el cual funciona como planificador de trabajos y monitor para la gestión proactiva de servidores distribuidos.

Con SQL Server podemos realizar operaciones remotas mediante un agente inteligente, el cual incluye un dispositivo evento-alerta-respuesta.

Microsoft SQL Server nos permite un chequeo de dispositivos de base de datos con un detector automático de fallos para la tolerancia a problemas con algunos dispositivos, realiza copias de seguridad que hayan sido desatendidas garantizando con ello la consistencia de datos para la más alta disponibilidad. Con los contextos de usuario protegidos, se pueden aislar los fallos de un único usuario. La recuperación point in time sirve para restaurar bases de datos o la transacción de logs en un intervalo de tiempo. Por el lado de la tolerancia a fallos de servidor, SQL Server permite un failover automático a un servidor de Backup (respaldo) o en espera.

Por el lado de la seguridad SQL Server utiliza un solo login tanto para la red como para la base de datos para mejorar la seguridad y facilitar la administración, tiene un Password (contraseña) y encripta los datos en red para mejorar la seguridad, la encriptación de los procedimientos nos permite tener mayor integración y seguridad de

código de aplicación. Existe interoperabilidad e integración con escritorios. Otro aspecto es que soporta totalmente API estándar DB Library y estándar ODBC nivel 2 como API nativa. También soporta importantes estándares de mercado como ANSI SQL-92, FIPS 127-2, XA y SNMP.

Por último a continuación resumiremos las características más relevantes de SQL Server 2000.

#### INNOVACIÓN Y LIDERAZGO

- Primera base de datos que se amplía desde los portátiles a la empresa mediante el mismo código base y que ofrece una compatibilidad del código del cien por cien.
- Primera base de datos que soporta la configuración automática y la auto-optimización.
- Primera base de datos con un servidor OLAP integrado.
- Primera base de datos con los servicios de transformación de datos (*Data Transformation Services*, DTS) integrados.
- El marco de almacenamiento de datos de Microsoft (*Data Warehousing Framework*) constituye el primer planteamiento de amplia cobertura para la resolución de los problemas que plantea la utilización de metadatos.
- La primera base de datos que ofrece administración multiservidor para un gran número de servidores.
- Una gran variedad de opciones de duplicación de cualquier base de datos.
- La mejor integración con la familia Windows NT Server, Microsoft Office y BackOffice®.
- Acceso universal a los datos (*Universal Data Access*), la estrategia de Microsoft para permitir el acceso de alto rendimiento a una gran cantidad de fuentes de información.

#### FACILIDAD DE USO

- Proporciona a los desarrolladores un modelo de programación simple y rápido, eliminar la necesidad de administrar la base de datos en las operaciones habituales y proporciona herramientas sofisticadas para acometer las operaciones más complejas.
- SQL Server 2000 reduce el costo total de propiedad mediante opciones tales como la administración de varios servidores con una única consola; ejecución de trabajos basados en eventos y generación de alertas; seguridad integrada y procedimientos de comandos para realizar tareas administrativas. Esta versión también deja vía libre al administrador de la base de datos para llevar a cabo trabajos más sofisticados al automatizar las tareas rutinarias. Mediante la combinación de estas potentes utilidades para la administración con las nuevas opciones de configuración automática, Microsoft

SQL Server 2000 constituye la opción ideal para las aplicaciones de automatización de sucursales y de bases de datos incrustadas

#### AMPLIABLE Y FIABLE

- Los clientes invierten en sistemas de administración de bases de datos en forma de aplicaciones escritas para sus bases de datos y también en la formación que conlleva su administración y despliegue. Esta inversión debe estar protegida: a medida que el negocio crece, la base de datos debe crecer para tratar más datos, transacciones y usuarios. Los clientes también desean proteger su inversión cuando llevan las aplicaciones de base de datos a equipos portátiles o a sucursales.
- Para satisfacer estas necesidades, Microsoft ofrece un único motor de base de datos ampliable desde un equipo portátil que ejecuta el sistema operativo Windows® 95 o Windows 98, hasta clusters multiprocesador simétricos de varios terabytes de información y que ejecutan Windows 2000 Server Enterprise Edition. Todos estos sistemas mantienen la seguridad y fiabilidad que exigen los sistemas empresariales críticos.
- Una novedad de la versión 2000 es su diseño para cubrir las necesidades cada vez mayores del mercado de la informática móvil, con nuevas e innovadoras funcionalidades como un pequeño espacio físico para la memoria, ajuste automático y duplicación en varias instalaciones.
- SQL Server es también la elección ideal para los sistemas de almacenamiento de datos y OLTP de la gama alta, ya que dispone de funcionalidades de escalabilidad como bloqueo dinámico a nivel de filas, paralelismo entre consultas, consulta distribuida y mejoras en bases de datos de gran tamaño (*Very Large Database, VLDB*).

#### ALMACENES DE DATOS

- Los sistemas de proceso de transacciones siguen siendo un componente fundamental de las infraestructuras de bases de datos corporativas. Las empresas también realizan grandes inversiones en mejorar el conocimiento de sus datos. La estrategia de Microsoft consiste en reducir el coste y la complejidad del almacenamiento de datos al tiempo que pone la tecnología al alcance de un mayor número de personas.
- Microsoft ha establecido un planteamiento de amplia cobertura para el *proceso* completo del almacenamiento de datos. El objetivo es facilitar aún más la creación y el diseño de soluciones de soluciones económicas de almacenamiento de datos mediante la combinación de tecnologías, servicios y alianzas entre fabricantes.
- La *Microsoft Alliance for Data Warehousing* es una coalición que reúne a los líderes del sector en el almacenamiento de datos y aplicaciones. El marco de almacenamiento de datos de Microsoft (*Microsoft Data Warehousing Framework*) es un conjunto de interfaces de programación que ha sido diseñado para simplificar la integración y administración de soluciones de almacenamiento de datos.
- Entre las innovaciones que se incluyen en SQL Server 2000 destinadas a mejorar el

- proceso de almacenamiento de grandes cantidades de datos, se encuentran:
- "Plato", un componente primordial para las soluciones empresariales que requieran proceso analítico en línea (*Online Analytical Processing, OLAP*), desde la generación de informes y análisis corporativos hasta el modelado de datos y el soporte en la toma de decisiones.
  - *Data Transformation Services* (Servicios de transformación de datos) para importar, exportar y transformar datos.
  - Mejoras en el tratamiento de las consultas complejas y bases de datos de gran tamaño (VLDB).
  - *Microsoft Repository* (Depósito de Microsoft), una infraestructura común para compartir la información.
  - Herramientas visuales de diseño para crear y mantener los diagramas de bases de datos.
  - Duplicación integrada, que incluye la actualización en varias instalaciones, para mantener almacenes de datos dependientes.
  - Integración de soluciones de terceros.

En la figuras 2.4.1 y 2.4.2 podemos ver la consola que nos ofrece SQL Server 2000 para la administración de las bases de datos.

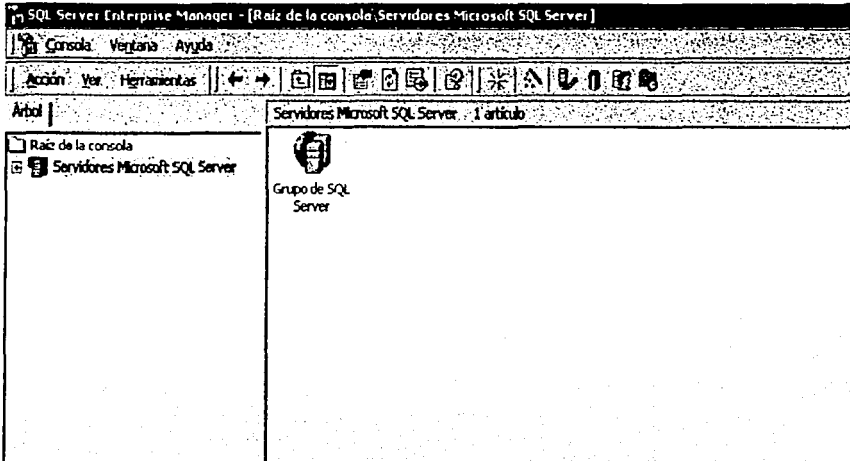


Figura 2.4.1 Raíz de la consola de SQL Server 2000



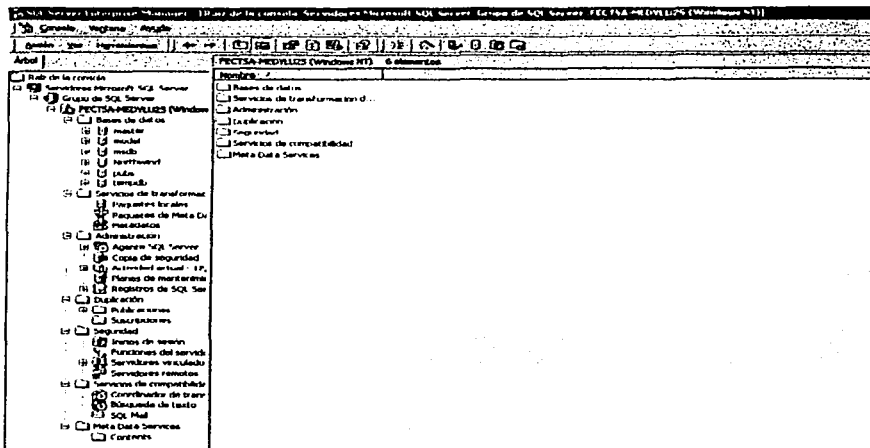


Figura 2.4.2 Bases de Datos

Las ventajas del SQL Server son la asignación dinámica de recursos, la cual permite la escalabilidad del uso del disco y memoria para adecuarse a las necesidades de la base de datos en cada momento. Esta característica de flexibilidad permite un mejor rendimiento y simplifica la administración del software, otra ventaja es que se pueden eliminar dispositivos.

El soporte para las plataformas de Windows versión 9x aumenta de manera importante la base de aplicaciones posibles para el SQL Server. El analizador gráfico de consultas es una herramienta útil y a menudo necesaria para construir y ejecutar comandos interactivos de SQL. El nuevo analizador de sentencias representa un paso adelante, debido a que no sólo se pueden construir unos procedimientos guardados y ejecutar consultas interactivas, sino que también se puede enseñar gráficamente los pasos que el procesador de consultas utiliza para ejecutar las misma

Los servicios de transformación de datos, la nueva característica de este producto es una poderosa herramienta, que nos permite facilitar el almacenamiento de los datos, además de simplificar la importación y exportación de los datos entre dos bases de datos compatibles con OLE DB, también genera scripts en Visual Basic los cuales se pueden ejecutar desde el Windows Scripting Host u otros entornos COM (Component Object Model). Las funciones del Enterprise Manager nos permite observar los contenidos de una tabla directamente desde el EM, además de que nos brinda la posibilidad de cambiar directamente los tipos de datos de las tablas existentes.

En cuanto a las desventajas de Microsoft SQL Server la instalación de este paquete requiere del Internet Explorer 4.0 o superior, lo cual no nos proporciona ninguna utilidad debido a que no tenemos un navegador de Web en nuestro servidor de la base de datos, y su instalación es un problema ya de por sí. Para llevar a cabo una migración es necesario un reinicio de la base de datos, lo cual implica que tal vez experimentemos una posible pérdida de datos.

## 2.5 Características, ventajas y desventajas de Powerbuilder 7.0

PowerBuilder es una herramienta de desarrollo de Aplicaciones Orientadas a Objetos en ambiente gráfico; que permiten construir diferentes tipos de aplicaciones que corran en diversas plataformas y que pueden acceder a varias Bases de Datos. PowerBuilder nos proporciona todas las herramientas necesarias que nos permiten desarrollar aplicaciones en Cliente/servidor, aplicaciones distribuidas y aplicaciones en Internet; permitiendo así un desarrollo seguro y robusto; fácilmente utilizado, tanto en pequeñas empresas como grandes, para construir desde una aplicación de contabilidad hasta un sistema completo de manufactura.

La interfaz de usuario de una aplicación en PowerBuilder, consiste de menús, controles de ventanas y ventanas que interactúan con los usuarios. Las aplicaciones construidas pueden incluir todos los controles de ventanas estándar tales como botones, menús, listas y controles de edición, entre otras. Lo que hace que PowerBuilder sea una herramienta fácil de desarrollar y muy amigable en su uso.

El proceso lógico de la aplicación se realiza a través de Eventos y funciones Scripts con las cuales se pueden codificar reglas de negocio, validación de reglas, flujos de proceso y actualizaciones a Base de Datos. Las ventanas, controles y otros componentes de la aplicación pueden tener cada uno, un conjunto de eventos pre-definidos, que mediante el uso del Lenguaje Scripts permiten a través de comandos, funciones y sentencias, programar una acción en respuesta a un evento.

Cada ventana o menú creado con PowerBuilder está por sí mismo contenido en un módulo llamado objeto. Cada objeto contiene características particulares y necesarias como: propiedades, eventos y funciones, que son propias del objeto. Además, pueden usar las técnicas avanzadas de programación orientada a objetos, tales como encapsulamiento, herencia y polimorfismo al crear los objetos, a través de interfaces

propias del usuario o módulos separados llamados **custom class user** (clases propias de usuario).

La figura 2.5.1 muestra una ventana del ambiente de desarrollo de PowerBuilder, en la cual podemos ver a través del visualizador diversos objetos. A continuación describiremos algunos de los términos básicos que aparecen en ésta figura:

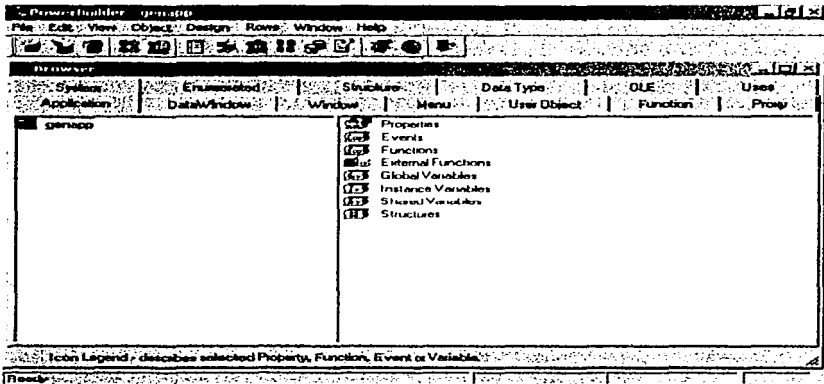


Figura 2.5.1 Visualizador de Objetos de PowerBuilder

**Aplicaciones:** Una aplicación es una colección de ventanas de PowerBuilder que realizan ciertas actividades, tales como una orden de entrada o actividades de contabilidad. El objeto de una aplicación es el punto de entrada dentro de Ventanas y controles que realizan estas actividades.

**Objeto:** En PowerBuilder, es una identidad gráfica que puede ser guardada en una librería (una aplicación, un DataWindow, Ventana, Menú u objeto de usuario). Pero algunos objetos pueden no ser visuales, (como una Transacción, un objeto definido por el usuario o una estructura). Cada objeto tiene un conjunto de propiedades que describen su apariencia y conducta. Se pueden usar funciones y sentencias o asignar

un operador de prueba y un conjunto de propiedades de un objeto. La definición de los objetos esta almacenada en una biblioteca llamada CLASE.

**Windows:** Son las Ventanas tradicionales, utilizadas en cualquier lenguaje de programación orientada a objetos.

**DataWindows:** Es una combinación de datos, operadores y funciones que generan un valor. Un Datawindow puede incluir un resultado de un Query, con un formato preestablecido, además de textos y un formato de ventana.

**Queries :** Son sentencias de SQL para realizar consultas o acciones sobre una Bases de Datos.

**Estructuras:** Es una colección de una o más variables relacionada, de un mismo tipo o diferente, que están agrupadas bajo un único nombre.

**Eventos:** En PowerBuilder hay diversos tipos de eventos los cuales se describen en la tabla 2.5.2

Tipo de Evento	Respuesta
Eventos de sistema con ID	Una acción de usuario o mensajes de otro sistema o a llamados en scripts
Eventos del sistema sin ID	Regularmente con mensajes de PowerBuilder o llamadas en scripts
Eventos predefinidos por el usuario con ID	Acción del usuario o un mensaje del sistema.
Eventos definidos por el usuario sin un ID	En llamadas en scripts

Tabla 2.5.2. Tipos de Eventos de PowerBuilder.

---

## Acceso a Bases de Datos.

Hablando en términos de acceso a Bases de datos, PowerBuilder nos proporciona un acceso fácil a la información almacenada en una gran variedad de Bases de Datos, utilizando tres tipos de accesos: ODBC<sup>30</sup>, JDBC<sup>31</sup> y OLE.

La Interfaz ODBC permite acceder a la Base de Datos mediante la conexión estándar de Microsoft. Cuando se utiliza esta interfaz, se define el ODCB de origen de los datos, el cual consiste de un conjunto de datos al que se quiere acceder y esta asociado a una DBMS<sup>32</sup> o un manejador de archivos, sistema operativo y si se presenta, a una red de software que accede a un DBMS. Con un ODCB, tus datos pueden residir localmente o remotamente en un servidor de red. Mediante la interfaz ODCB se puede definir: (1) Una biblioteca de función ODBC de llamadas que conecta la base origen, ejecutan sentencias de SQL<sup>33</sup> y recuperan datos; (2) una conexión estándar y un **logon** (entrada) a DBMS; (3) el uso de la sintaxis SQL; (4) una representación estándar de los tipos de datos y; (5) el manejo de errores.

Las aplicaciones que proporcionan una interfaz ODBC, como PowerBuilder, pueden acceder a fuentes de datos para cualquier manejador ODBC existente. EL manejador ODBC es una librería de enlace dinámico o una librería compartida que implementa las funciones de llamadas del ODBC. PowerBuilder utiliza una librería PBODB070.DLL, para establecer una conexión con la Bases de Datos.

La Interfaz JDBC, es una interfaz de una aplicación estándar (API) que permite a una aplicación JAVA acceder a cualquier Base de datos que soporte SQL como lenguaje estándar de acceso. El API de JDBC incluye clases para actividades de Bases de datos, tales como conexiones abiertas, ejecución de comandos propios de SQL y proceso de resultados. Mediante la interfaz JDBC se define: (1) Una biblioteca de función JDBC de llamadas que conecta la base origen, ejecutan sentencias de SQL y recuperan datos; (2) una conexión estándar y un **logon** (entrada) a DBMS; (3) el uso

---

<sup>30</sup> ODBC , Conexión abierta de Bases de Datos

<sup>31</sup> JDBC Conexión a la Base JAVA.

<sup>32</sup> DBMS, Sistema de Administración de Base de Datos.

<sup>33</sup> SQL Lenguaje estructurado de Consultas.

de la sintaxis SQL; (4) una representación estándar de los tipos de datos y; (5) el manejo de errores. PowerBuilder soporta 2 versiones de Interfaz JDBC: JDM la cual corre para Microsoft y JDS para SUN.

Cada Interfaz JDBC tiene su propia interfaz DLL<sup>34</sup> que se comunica con una Base de datos en específico. Cada Interfaz de Base de Datos JDBC usa su propia interfaz DLL para comunicarse con una base de datos específica. Esto es porque cada DLL debe conectarse a una Base de Datos a través de diferente API<sup>35</sup>. En contraste con la ODBC que usa una simple API para comunicarse con la Base de Datos. Una interfaz JDBC no utiliza un driver para comunicarse con la Base por lo cual es necesario crear un **profile** (perfil de usuario) con la información específica que PowerBuilder necesita para conectarse a la Base de Datos.

La interfaz OLE DB API, permite a las aplicaciones acceder a gran variedad de datos para cualquier dato OLE DB proporcionado que exista. Ésta proporciona una aplicación con un acceso uniforme a los datos almacenados en diversos formatos tales como archivos secuenciales indexados, Bases de Datos, herramientas productivas y DBMS basados en SQL.

## VENTAJAS

- Una de las más importantes ventajas del PowerBuilder es su fácil conexión a gran variedad de bases de datos, a través de una conexión ODBC, JDBC y OLE DB. Mediante estas conexiones PowerBuilder puede conectarse a Bases de Datos IBM (DB2 e Informix, ) a Bases de Datos de Microsoft SQL Server 6 en adelante, Oracle y por supuesto Sybase.

- PowerBuilder soporta cualquier plataforma de desarrollo y de trabajo, por ejemplo se puede desarrollar aplicaciones usando PowerBuilder en Windows y trabajar la misma aplicación con UNIX o viceversa, también se puede tener un equipo de

---

<sup>34</sup> DLL Librerías de Enlace Dinámico

desarrolladores usando Windows y algunos usando UNIX que desarrollen la misma aplicación al mismo tiempo.

La tabla 2.5.3 muestra las características técnicas más sobresalientes de PowerBuilder.

Producto incluidos en PowerBuilder.	Windows (empresarial)	UNIX (empresarial)
Ambiente de Desarrollo Interactivo (IDE)	Disponible	Disponible
Conexión mediante ODBC.	Acceso Completo	Acceso Completo
Uso del Servidor Adaptive Anywhere para desarrollo.	Disponible	Disponible
Biblioteca de Clases.	Disponible	Disponible
SCC Interfaz para control de Fuentes	Disponible	No Disponible
Uso de procedimientos almacenado.	Disponible	Disponible
Conexión OLE DB	Disponible	No Disponible
Conexión mediante controladores DBMS	Disponible	Disponible
Componentes de desarrollo y despliegue para Jaguar CTS.	Disponible	Disponible
Componentes de desarrollo y despliegue para COM/MTS.	Disponible	No Disponible
Conjunto completo de PowerSite, incluyendo el HTML DataWindow DTC	Disponible	No Disponible
Nueva HTML DataWindow	Disponible	Disponible
InfoMaker	Disponible	No Disponible

**Tabla 2.5.3. Productos incluidos en PowerBuilder**





## **CAPÍTULO III PLANTEAMIENTO DEL PROBLEMA Y PROPUESTA DE SOLUCIÓN**

### **3.1 Problemática Actual**

El presente sistema controla la parte de cobranza de una empresa dedicada al préstamo hipotecario, la problemática actual que esta área presenta es la siguiente:

Debido a que el proceso de cobranza se lleva a cabo actualmente de forma manual se consideró como proyecto prioritario la elaboración de un sistema que automatice dicho proceso y que sea capaz, entre otras cosas, de controlar y agilizar el cobro para contar con información actualizada y tener recursos activos disponibles. Para lograr esto, se hará uso de las tecnologías de informática y comunicaciones con las que cuenta la empresa.

Actualmente la empresa posee un sistema de administración de bases de datos en SQL Server 2000 y como interfaz la herramienta de Power Builder, lo cual se ajusta a los requerimientos de nuestra propuesta.

Algunas de las situaciones que actualmente se presentan fueron consideradas en el planteamiento del nuevo sistema, tomando en cuenta que el pago se lleva a cabo de forma manual, los inconvenientes que surgen son los siguientes:

- Inversión de tiempo al hacer las validaciones y cálculos manualmente.
- Papeleo innecesario y trasapeleo.
- Retardo en el cobro, debido a que la información no está disponible en todo momento.
- Mayor posibilidad de cometer errores al hacer cálculos o validaciones.
- No se puede verificar de forma inmediata los depósitos de los cheques.

- No se puede exigir el pago inmediato de intereses moratorios al no tener la información disponible en el momento.
- El proceso de consulta es muy tardado.
- Los documentos recibidos salvo buen cobro no pueden ser dispuestos el mismo día del depósito, por lo que en caso de ser rechazados, aparecerán en línea empresarial como cargos y no como depósitos.
- No se cuenta oportunamente con la información necesaria para agilizar trámites:
  - Cheques devueltos
  - Motivo de rechazo
  - Pagos en tiempo
  - Pagos en mora
  - Intereses que no han sido cubiertos

Todo lo anterior hace necesario implantar un sistema que agilice el proceso, que sea sencillo y fiable para el usuario, y que sus limitaciones correspondan al perfil establecido por su actividad dentro del área en la cual el sistema será implantado.

La aplicación deberá mantener comunicación con los catálogos establecidos por la empresa y se enlazará con una base de datos que contendrá la información necesaria.

Además, la implantación del sistema permitirá contar con la información en línea, lo cual hará posible verificar los depósitos de los cheques en forma inmediata y hacer una validación contra los depósitos en el banco, ya que estarán en un sistema centralizado.

En la siguiente figura presentamos un diagrama que muestra la situación actual del proceso de la información, desde la recepción de los documentos de pago hasta el cálculo de intereses moratorios.

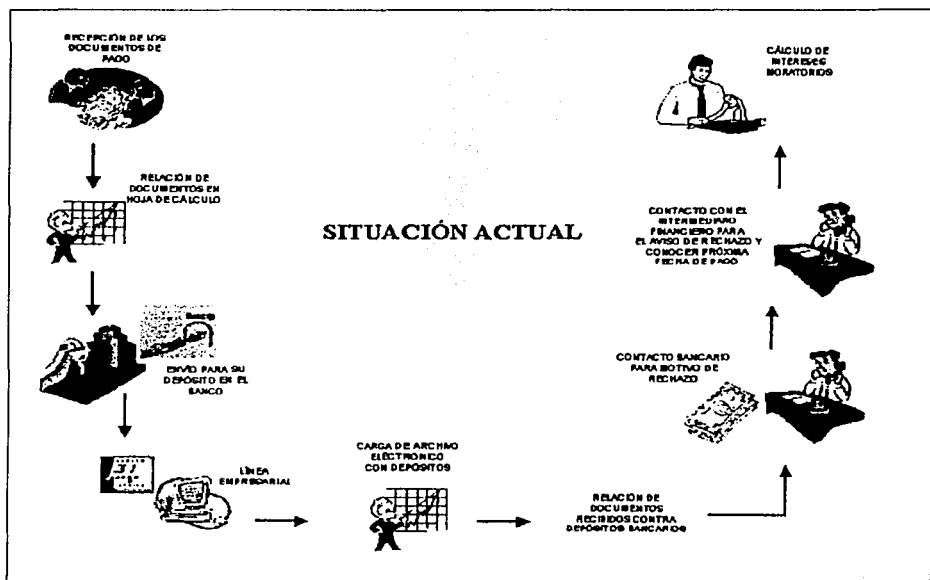


Figura 3.1.1 Proceso de cobranza actual de una empresa hipotecaria.

### 3.2 Requerimientos de la empresa.

Ante el problema, antes expuesto, en el área de cobranza; la empresa hipotecaria desea agilizar sus procesos a través de un sistema que cumpla con los siguientes requerimientos:

- Seguridad.

En este aspecto, la empresa solicita que el acceso a la ejecución de una opción dentro del menú dependerá de los permisos que se asignen a los usuarios, previa autorización del jefe inmediato. Por lo tanto, los usuarios no autorizados para acceder a ciertas opciones del menú, tendrán deshabilitadas dichas opciones al ingresar a la aplicación. Y sólo será el jefe del departamento quién podrá acceder por default a todas las opciones del sistema.

- Operación rápida y eficiente.

Esto se refiere al procesamiento de la información, la cual se desea sea mediante una interfaz sencilla y amigable para el usuario final.

- La aplicación deberá ejecutarse desde una computadora con sistema operativo windows95 o posterior.

En este punto, también conviene señalar, que se requiere que el cliente del manejador de bases de datos relacional esté instalado en dicha computadora para establecer conexión a la base de datos.

- Usar un manejador de bases de datos relacional.

Se solicita un manejador de bases de datos relacional debido a la simplicidad y generalidad que ofrecen en el manejo de datos.

- Deberá, también, contar con manuales del usuario y técnico.
- El sistema deberá ser gráfico.

La empresa ha hecho énfasis en la importancia de seguir los estándares preestablecidos tanto en la codificación como en la presentación de los objetos del frontend, los cuales se detallan a continuación.

### **3.2.1 Estandarización y Convenciones de Codificación**

#### **Estructuración del proyecto**

Para almacenar toda la información relacionada con el proyecto se contará con un subdirectorio especial (Sistema) que incluirá solamente archivos y subdirectorios propios del proyecto. Dentro de este directorio existirán los subdirectorios correspondientes a cada módulo y dentro de cada subdirectorio se deberá crear otro para guardar los archivos correspondientes como se muestra en el esquema de la figura 3.2.1.1.

Las librerías no deberán exceder los 800 Kb de tamaño ó 70 objetos, en caso de que la librería crezca demasiado y rebase los límites establecidos, se deberá generar otra librería con el nombre correspondiente y un número consecutivo.

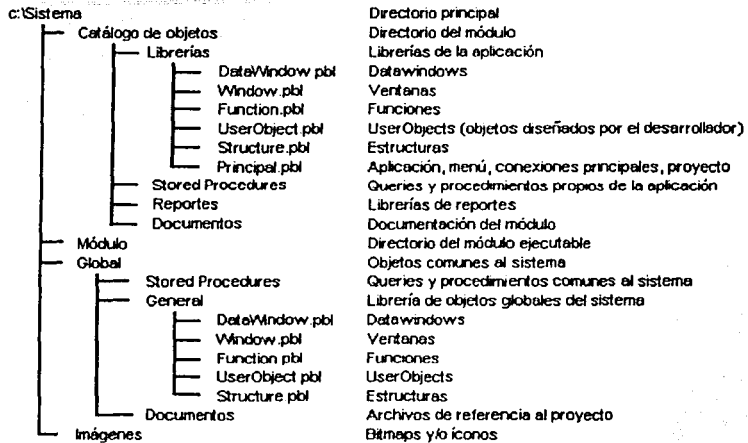


Figura 3.2.1.1 Esquema de organización de archivos del sistema.

### Creación del ejecutable

Los archivos ejecutables (\*.exe) deberán guardarse en el subdirectorio c:\Sistema\Módulo.

### Estándares para el tipo de ventana

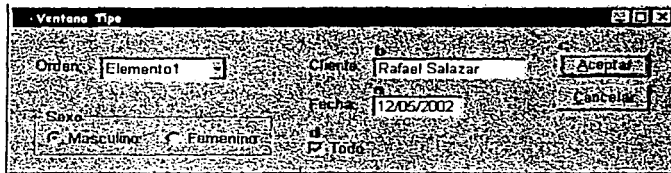


Figura 3.2.1.2 Ventana tipo

Identificador	Descripción	Separación
<b>A</b>	Espacio vertical con la misma funcionalidad	<b>27 upb*</b>
<b>B</b>	Espacio horizontal con la misma funcionalidad	<b>40 upb</b>
<b>C</b>	Espacio vertical con diferente funcionalidad	<b>55 upb</b>
<b>D</b>	Espacio horizontal con diferente funcionalidad	<b>90 upb</b>

\* Unidades de Power Builder

**Figura 3.2.1.3 Descripción del espaciado entre controles**

Control	Alto	Ancho	Tipo	ID
Static text	53	Requerido, según dato	n/a	st
DropDownListBox	253	Requerido, según dato	n/a	ddl
SingleLineEdit	77	Requerido, según dato	n/a	sle
EditMask	77	302	Fecha	em
EditMask	77	302	Hora	em
EditMask	77	513	Fecha-hora	em
EditMask	77	Requerido, según dato	Numérico o Cadena	em
CommmanButton	97	353	n/a	cb
CheckBox	53	Requerido, según dato	n/a	cbx
RadioButton	53	Requerido, según dato	n/a	rb
PictureButton	Requerido, según dato	Requerido, según dato	n/a	pb
DataWindow	Requerido, según dato	Requerido, según dato	n/a	dw
Picture	Requerido, según dato	Requerido, según dato	n/a	p
GroupBox	Requerido, según dato	Requerido, según dato	n/a	gb

**Figura 3.2.1.4 Estándares según el tipo de objeto**



---

### Características generales:

- El tipo de letra a utilizar en los textos de los controles debe ser Arial Normal con un tamaño de 10, utilizando siempre Mayúsculas y Minúsculas.
- El tipo de letra a utilizar en los reportes con **DataWindows** (ventana de datos) debe ser Arial Normal con un tamaño de 10 u 8 (dependiendo de la cantidad de información), utilizando siempre Mayúsculas y Minúsculas, procurando que los datos sean claros y el tamaño de los campos ajuste el largo de los textos desplegados.
- Cuando la etiqueta esté relacionada con un control de edición, el texto de ésta deberá terminar con “.”.
- Ajustar el ancho de las etiquetas que tengan relación, considerando la etiqueta de mayor tamaño, para que éste sea el mismo en todas.
- No olvidar asignar aceleradores a las etiquetas y asociar éstos a los controles respectivos.
- Los **CommandButtons** (botones de comandos) de Aceptar y Cancelar no tendrán aceleradores, pero estarán asociados a los atributos de Default y Cancel respectivamente.
- En los CommandButtons cuya operación sea la apertura de una ventana, el texto deberá terminar con puntos suspensivos “...”.
- El borde a utilizar para aquellos controles que lo requieran será de 3D lowered.
- El espacio para el primer control de la ventana con respecto a la orilla será a la izquierda de 55 upb y como tope 37 upb.
- Utilizar las opciones de ajuste de tamaño, espaciado y alineaciones que PowerBuilder ofrece para facilitar el diseño de la pantalla.
- Los controles que servirán de base para realizar espaciados o alineaciones en grupo serán: sle, em, dd1b y cb.
- La alineación vertical de la etiqueta con respecto a su control de edición, deberá estar centrada.
- Asignar el **TabOrder** (orden del tabulador) correspondiente para definir el orden de navegación de la ventana.

- Para el despliegado de mensajes se utilizará la instrucción `MessageBox`, en la que deberá indicarse una de las opciones de títulos y uno de los íconos que se describen en la tabla de la figura 3.2.1.5.

Título	Ícono
Dar información al usuario	Information! (Default)
Detener el proceso e informar el motivo	StopSign!
Dar aviso de un proceso con problemas que genera consecuencias	Exclamation!
Preguntar al usuario para toma de decisiones	Question!

Figura 3.2.1.5 Título e íconos del `MessageBox`

### Documentación de los objetos

Cada objeto deberá contener, al inicio de cada `script` (código) el comentario que se muestra a continuación, indicando la funcionalidad del mismo.

```
//*****  
// Nombre del objeto:  
// Nombre del evento:  
// Observaciones: (Aspectos importantes que deben ser tomadas en caso  
//                 de futuras modificaciones)  
// Desarrollo:    (Quién)                               Fecha:    (Cuando se creó)  
// Modificaciones: (Nombre del programador, fecha y descripción de la modificación)  
//*****  
// Declaración de constantes  
// Declaración de variables  
// Verificación  
// Procesos
```

### Documentación de funciones

Todas las funciones deberán iniciar con los comentarios indicados abajo, cuyo objetivo es el describir claramente su función. El programador debe sustituir el texto cursivo con el texto apropiado. Es importante que el desarrollador revise y mantenga actualizado el

comentario cuando hace modificaciones sobre una función ya que ésta servirá de punto de partida para la generación de los manuales de referencia.

```
//*****  
// Nombre de la función:  
// Descripción: (Funcional)  
// Declaración: (Cómo está declarada la función)  
// Parámetros: (Para qué son utilizados)  
// Observaciones: (Aspectos importantes que deben ser tomadas en cuenta en caso  
// de futuras modificaciones)  
// Regresa valores: (Tipo de variables y descripción)  
// Desarrollo: (Quien) Fecha: (Cuando se creó)  
// Modificaciones: (Nombre del programador, fecha y descripción de la modificación)  
//*****  
// Declaración de constantes  
// Declaración de variables  
// Verificación  
// Procesos
```

### Comentarios en el código.

Se deben incluir comentarios dentro del código con el fin de tener ayuda extra en el mantenimiento de los programas. No se pretende que cada línea sea comentada, lo cual resultaría contraproducente, sino indicar comentarios por bloques, explicando el por qué se hace, sobre todo cuando el propio código es complicado por sí mismo.

Otra cuestión importante es marcar, por medio de comentarios cortos, cuando cierta parte del código depende de objetos externos como tablas, triggers, stored procedures, otras ventanas u otros.

Se debe documentar, lo más claro posible, cada uno de los procesos. En el caso de actualizaciones, se debe indicar el nombre de la persona y la fecha en el encabezado de cada objeto.

### Comentarios en los objetos

Todos los objetos deben comentarse indicando el objeto, tipo de objeto y módulo al que pertenecen.

Ej.     w\_child\_clientes            Ventana **child** (hija) de clientes módulo 1.  
       dw\_tab\_clientes            Datawindow tabular de clientes módulo 1

### Especificaciones de formatos y alineaciones

Campos	Formato	Alineación
String (carácter) variable		Izquierda
String fijo		Centrado
Número	###,###,##0	Derecha
Money (moneda)	\$ ###,###,###0.00	Derecha
Porcentaje	##0.00 %	Derecha
Fecha	dd/mm/yyyy	Centrado
Hora	hh:mm:ss	Centrado

### Especificaciones de reportes

El cliente debe definir el estándar desde el primer reporte, según los objetos que se muestran en la figura 3.2.1.6.

- Tipo de letra y tamaño en el título, encabezados, contexto.
- Espacios entre objetos.
- Longitud y ancho de cada objeto.
- Formato de cada campo.
- Alineación de cada objeto.

Fecha Páginas: 1 de 1

LOGO (opcional) Empresa

Título del reporte

Criterios de selección

Campo1	Campo2	Campo3
--------	--------	--------

Grupos:

999	Descripción	###,###
999	Descripción	###,###

---

<usuario> <w\_reporte>

Figura 3.2.1.6 Propuesta de formato de reporte

### Interfaz y funcionalidad de la aplicación

La interfaz se refiere a la apariencia física que tienen los objetos visuales que aparecen en pantalla: sus dimensiones, colores, apariencia, separaciones, vistas tridimensionales, entre otros.

La funcionalidad se refiere a la forma que tiene el usuario de interactuar con los objetos visuales y las relaciones que existen entre ellos.

Juntos, interfaz y funcionalidad, conforman la operatividad de la aplicación. Éstas deben ser definidas en el prototipo inicial, tomando en cuenta el diseño visual (organizar la información de forma más clara), diseño conceptual (mecanismo de operación, cómo trabaja) y las necesidades reales de operación del usuario. En la figura 3.2.1.7 se plantea la forma operativa y funcional de una ventana.

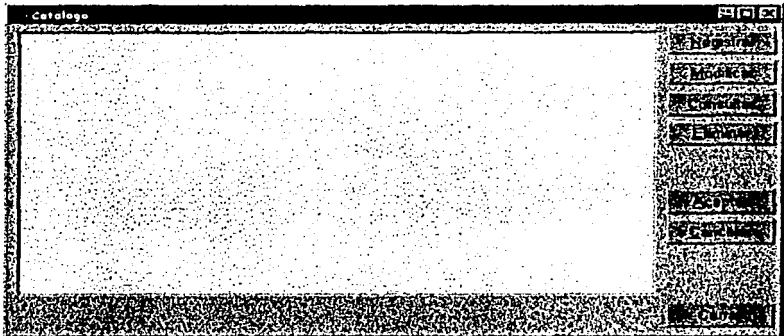


Figura 3.2.1.7 Ventana de catálogo

### Convenciones para la creación de un menú

El menú es el encargado de dar la pauta para la ejecución de las ventanas, desde él son disparadas las acciones, sin que ello signifique que sólo la ventana puede ser programada. En la figura 3.2.1.8 podemos observar el orden esquemático recomendado para organizar el menú de una aplicación.

- La ventana principal debe tener su menú.
- Se debe incluir el menos los submenús "Archivo", "Ventana" y "Ayuda".
- Toda acción se debe escribir con la primera letra en mayúscula.
- Toda acción del menú debe tener su tecla de acceso rápido (ALT + letra).
- Las acciones más usadas deben tener un acelerador asociado (CTRL + letra).
- Toda acción que involucre abrir una ventana, debe llevar la terminación "...".
- Se deben utilizar separadores para indicar grupos dentro de un submenú.
- Cada grupo no debe tener más de siete elementos.
- La estructura no debe pasar de 5 niveles.
- Si una acción es inaccesible en cierto momento, deberá ser deshabilitada.
- Se debe evitar el uso de más de 3 palabras para describir la acción.
- Se deben utilizar las mismas palabras para describir los mismos conceptos.

- El nombre del título del menú deberá ser igual al título de la ventana que abre.

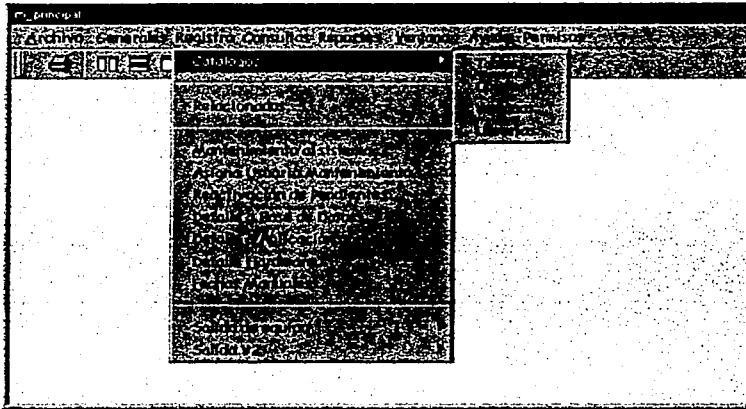


Figura 3.2.1.-8 Organización recomendada para un menú

Es importante no mezclar acciones que no tienen una relación lógica dentro del mismo submenú. Por otra parte, el menú siempre debe representar el estado del sistema. Si una acción no se puede ejecutar, es inaccesible o no aplica en cierto momento, ésta debe ser deshabilitada del menú. Si una opción del menú representa un estado lógico, es decir, representa un valor **booleano** (si o no), éste debe indicarse con la presencia de una palomita en la opción (menú checked).

### Estandarización en la nomenclatura

De igual manera PowerBuilder, por default, ofrece un prefijo para cada control que maneja, esto es muy importante, ya que por medio del mismo podemos identificar fácilmente qué tipos de controles estamos utilizando dentro de cada ventana. En la figura 3.2.1.9 se presenta una lista de cada control, acompañado del prefijo utilizado.

Control	Prefijo
CheckBox	cbx_
CommandButton	cb_
DataWindow	dw_
DropDownListBox	ddlb_
DropDownDataVWindow	dddw_
EditMask	em_
Graph	gr_
GroupBox	gb_
HScrollBar	hsb_
Line	ln_
ListBox	lb_
MultiLineEdit	mle_
Oval	oval_
Picture	p_
PictureButton	pb_
RadioButton	rb_
RoundRectangle	r_
Rectangle	rr_
SingleLineEdit	sle_
StaticText	st_
UserObject	uo_
VScrollBar	vsb_

Figura 3.2.1.9 Convenciones en la nomenclatura de los objetos

### Declaración de variables

El nombre de la variable debe estar compuesto de al menos dos elementos. Para separar cada elemento se debe utilizar **underscore** (guión bajo). El formato que debe utilizarse es:

<Prefijo>\_<Palabrasclave>

- Palabras clave: Son definidas por el usuario; deben ser representativas de la utilización de la variable sin incluir preposiciones ni conectores (y, de, para).
- Prefijo: Se compone de dos partes, la primera indica el tipo de variable (global, instancia o local) y la segunda parte indica el tipo de dato de la variable (entero, decimal, numérico, caracter).



Las figuras 3.2.1.10, 3.2.1.11 y 3.2.1.12 indican los estándares a seguir en el formato de las variables a utilizar en la codificación.

Tipo de Variable	Prefijo
Global	g
Instancia	i
Compartida	c
Local	l

Figura 3.2.1.10 Declaración de tipos de variables

Tipo de Función	Prefijo
Global	f_
Ventana	wf_
Menú	mf_
UserObject	uf_
Aplicación	af_

Figura 3.2.1.11 Declaración de funciones

Tipo de Dato	Prefijo
Boolean	b
Char	c
Date	dt
DateTime	dtl
Decimal	d
Double	db
Integer	i
Long	l
Real	r
String	s
SmallDateTime	sdt
SmallInteger	si
Time	t

Figura 3.2.1.12 Declaración del tipo de dato

### Indentación del código

La indentación del código debe ser por medio de Tabs con un valor de 3 espacios cada uno.

### Uso de estructuras de programación

#### *IF-THEN-ELSE*

En el uso de condiciones queda restringida su codificación en un solo renglón.

```
Incorrecto:      If Is_palabra = "SI" Then Is_palabra = "NO"  
Correcto:        If Is_palabra = "SI" Then  
                  Is_palabra = "NO"  
                  End If
```

#### *FOR-TO-NEXT*

Por claridad en el código se establece por convención que en el uso de ciclos se incluya el nombre de la variable en la cláusula NEXT de la instrucción. En el caso especial de tener ciclos FOR-NEXT anidados en los cuales se pretenda indexar arreglos, se permite el uso de variables tipo enteras con los nombres simples i, j, k.

```
Ejemplo:        For i = 1 To 10  
                  For j = 1 To 10  
                    Procesos  
                  Next j  
                Next i
```

### *El operador "&" vs. "+"*

El operador "+" denota, por naturaleza, una operación aritmética que no se parece en nada a una concatenación. Aún cuando se utilicen dos variables tipo string, con el operador "+" la situación puede ser algo confusa. Por tanto, como convención, el operador "&" siempre debe ser usado para las concatenaciones, donde no pueden aparecer más que strings y el operador "+" únicamente se debe utilizar para operaciones aritméticas, donde no aparecen más que tipos numéricos.

### *La propiedad por default*

Todo objeto tiene definida una propiedad por default que puede ser accesada por medio del nombre del objeto, sin hacer referencia al nombre de la propiedad con el separador "." Puesto que se pretende eliminar ambigüedades en el código, se ha convenido prohibir la omisión del nombre de la propiedad por default cuando se requiere hacer referencia a ella.

### *Abreviaturas*

El uso de abreviaturas en los nombramientos de entidades es muchas veces necesario para evitar los nombres muy largos. Algunas recomendaciones que se deben tomar en cuenta son las siguientes:

- Sólo abreviar palabras que son muy largas. No vale la pena sacrificar claridad por una o dos letras a suprimir en una palabra.
- Sólo abreviar palabras que son muy repetitivas a través del programa y donde no hay lugar a confusiones.
- Al crear abreviaturas, procurar que éstas sean fácilmente pronunciables.

- Si alguna palabra tiene su abreviatura definida en el diccionario o es una abreviatura de dominio público, tratar de usar esa y no inventar una nueva.
- Algunos nombres propios y ciertos conceptos que constan de varias palabras son, muchas veces, mejor identificados por sus siglas, en cuyo caso se deberán poner en mayúsculas.

### **3.3 Búsqueda y análisis de la información**

Tanto la obtención de la información como el análisis de la misma son parte fundamental del desarrollo de nuestro sistema, ya que los datos obtenidos nos facilitarán el diseño, desarrollo e implantación de una solución factible y exitosa. El objetivo de esta parte es comprender la situación y no resolver el problema, ya que la investigación y el cuestionamiento para conocer como opera el sistema actual de cobranza nos permite identificar los requerimientos de los usuarios para optimizar el sistema y proponer uno nuevo.

Después de comprender en su totalidad el sistema, estaremos en posición de cuestionarlo y generar recomendaciones, algo importante para esto es reunir la información apropiada, ya que esto va a representar la calidad de la aplicación.

La técnica para reunir dicha información fue la de entrevista no estructurada, ya que las ventajas que dicha técnica tiene son:

- El entrevistador tiene mayor flexibilidad para cambiar los términos de las preguntas para que se acomoden mejor al entrevistado. A diferencia de la entrevista estructurada en la cual se da un planteamiento mecánico.

- El entrevistador puede profundizar en áreas que le parezcan de mayor interés durante la entrevista o incluso en áreas que no habían sido tomadas en cuenta, en el caso de la entrevista estructurada esto no es posible.

Utilizando la técnica anterior nos fue posible obtener información sobre el flujo de información, las áreas que intervienen en el proceso, la descripción general de cada una de ellas y algunos de los formatos que se utilizan actualmente en el sistema de cobranza:

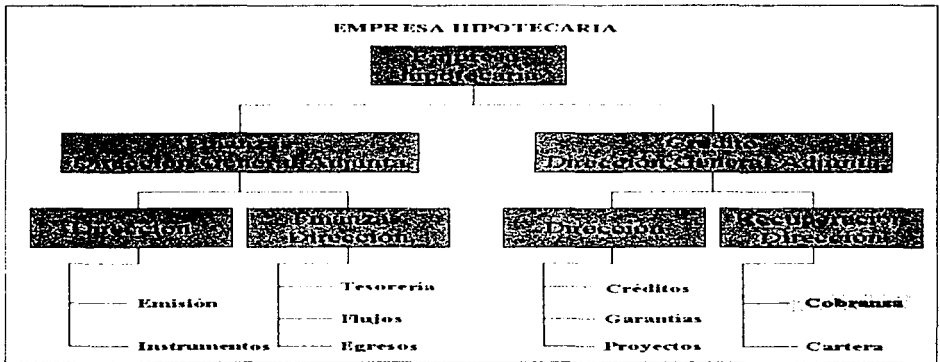


Figura 3.3.1 Organigrama de una empresa hipotecaria

TESIS CON  
FALLA DE ORIGEN

<b>DIRECCIÓN:</b>	<b>CATÁLOGO DE PUESTOS</b>
<b>OFICINA:</b>	<b>DIRECCIÓN GENERAL ADJUNTA DE FINANZAS</b>
<b>NOMBRE:</b>	<b>DIRECCIÓN DE FINANZAS</b>
<b>NÚMERO DE PLAZAS:</b>	<b>DIRECTOR</b>
<b>1</b>	
<b>ACTIVIDADES</b>	
<ul style="list-style-type: none"><li>• Apoyar a la Dirección General Adjunta de Finanzas.</li><li>• Coordinar las Subdirecciones de Tesorería, Flujos y Egresos.</li><li>• Proponer a la Dirección General Adjunta políticas de financiamiento, mecanismos crediticios y operativos.</li><li>• Participar en la definición y evaluación de las fuentes de financiamiento.</li><li>• Planear y promover las emisiones de títulos de la empresa.</li><li>• Analizar el riesgo de tasas de mercado.</li><li>• Analizar los riesgos de crédito.</li><li>• Establecer pautas de política a la Tesorería.</li><li>• Asistir a las reuniones de Comité.</li><li>• Coordinar, supervisar y controlar las actividades de las subdirecciones, así como las que le encomiende la Dirección General Adjunta.</li><li>• Estructurar y elaborar los presupuestos y calendario de Trabajo de la Dirección.</li><li>• Participar en la programación, seguimiento y evaluación de la gestión financiera de la Empresa y el seguimiento de la cobranza a los Intermediarios Financieros.</li></ul>	

Figura 3.3.2 Actividades correspondientes al área de Finanzas

<b>CATÁLOGO DE PUESTOS</b>	
<b>DIRECCIÓN:</b>	<b>DIRECCIÓN GENERAL ADJUNTA DE CRÉDITO</b>
<b>OFICINA:</b>	<b>DIRECCIÓN DE RECUPERACIÓN</b>
<b>NOMBRE:</b>	<b>DIRECTOR</b>
<b>NÚMERO DE PLAZAS:</b>	<b>1</b>
<b>ACTIVIDADES</b>	
<ul style="list-style-type: none"><li>• Planear, coordinar y supervisar las actividades a cargo de las subdirecciones.</li><li>• Elaborar y dar seguimiento a los programas relacionados con las actividades asignadas a las oficinas a su cargo, en coordinación con los titulares de las mismas.</li><li>• Establecer acuerdos con la Dirección General Adjunta respecto al desarrollo y atención de las actividades de las oficinas asignadas a su cargo.</li><li>• Supervisar el adecuado funcionamiento del sistema Integral de Información de la Empresa relacionado con las actividades de las oficinas asignadas a su cargo.</li><li>• Informar a la Dirección General Adjunta sobre el avance y cumplimiento de las metas establecidas en los programas.</li><li>• Diseñar y proponer el establecimiento de políticas y lineamientos para la operación y supervisión de los créditos que otorga la Empresa.</li><li>• Participar en el diseño de políticas de crédito.</li><li>• Diseñar los procedimientos de operación de los créditos.</li><li>• Supervisar el registro de proyectos de unidades habitacionales que financia la Empresa, así como las actividades orientadas a promover el desarrollo de tecnologías para la edificación de viviendas.</li><li>• Supervisar los créditos para la construcción e individualización de viviendas, así como los créditos de liquidez y las garantías por incumplimientos de los compradores.</li><li>• Supervisar las actividades de control de cartera y cobranza de los créditos otorgados por la Empresa.</li><li>• Coordinar las actividades de supervisión física de los proyectos financiados.</li><li>• Participar en la Integración y actualización de los manuales de operación, técnico y de supervisión de los créditos que otorga la Empresa.</li><li>• Supervisar la adecuada capacitación y desarrollo del personal asignado a la Dirección.</li><li>• Participar en los Comités que le designe la Dirección General de la Empresa.</li><li>• Coordinar con otras Direcciones el desarrollo de las actividades que se encuentren interrelacionadas.</li><li>• Elaborar los informes de operación de la Empresa para su presentación a la Dirección General y el Comité.</li></ul>	

Figura 3.3.3 Actividades correspondientes al área de Recuperación.

CATÁLOGO DE PUESTOS	
DIRECCIÓN:	DIRECCIÓN DE RECUPERACIÓN
OFICINA:	COBRANZA
NOMBRE:	SUBDIRECTOR
NÚMERO DE PLAZAS:	1
ACTIVIDADES	
<ul style="list-style-type: none"><li>• Coordinación de las actividades del personal del área de cobranza.</li><li>• Supervisar la correcta aplicación de intereses de acuerdo a las Condiciones Generales.</li><li>• Supervisar la recuperación oportuna de las erogaciones mensuales, pago de intereses y amortización de los créditos.</li><li>• Proponer a la Dirección las acciones necesarias para la recuperación de los créditos en cartera vencida.</li><li>• Establecer los controles necesarios para vigilar la correcta recuperación de la cartera de crédito y cobro de intereses.</li><li>• Dar seguimiento a la recuperación de los créditos no liquidados o con atraso en el pago.</li><li>• Supervisar el cobro de intereses moratorios en los casos en que los intermediarios no entreguen a la Empresa en los tiempos establecidos, los recursos de los créditos.</li><li>• Controlar el pago de intermediación financiera y el cobro de comisiones por intermediación en los créditos otorgados.</li><li>• Coordinar las actividades de los analistas relacionados con la actualización y mantenimiento de los saldos de cartera de los créditos otorgados.</li><li>• Atender las solicitudes de los intermediarios financieros para aclarar las diferencias entre los registros de cartera de la Empresa y los de dichos intermediarios.</li><li>• Elaborar y presentar a la Dirección los informes relacionados con la cartera de crédito.</li><li>• Participar en las reuniones relacionadas con las actividades de cobranza de los créditos.</li><li>• Participar en el diseño de los procedimientos para automatizar las actividades de control de cartera y cobranza de los créditos.</li><li>• Evaluación de las actividades del personal de la oficina.</li><li>• Proponer los programas de capacitación para el personal de la Oficina.</li><li>• Atender otras actividades asignadas por la Dirección.</li></ul>	

Figura 3.3.4 Actividades correspondientes al área de Cobranza



**PAPEL MEMBRETADO**

**FORMATO: AAA-001**

**Pago Cobranza**

**LICENCIADO**

Director General de la  
EMPRESA HIPOTECARIA  
Dirección  
Colonia  
Código postal, entidad federativa, país

Fecha: \_\_\_\_\_

De acuerdo con los contratos de apertura de crédito que tenemos celebrado con  
ustedes, anexo cheque núm. \_\_\_\_\_ a cargo de \_\_\_\_\_ por la  
cantidad de \$ \_\_\_\_\_ que cubre los intereses por el crédito otorgado de las  
erogaciones cobradas del mes de \_\_\_\_\_ según relación anexa o diskette anexo.

Atentamente

\_\_\_\_\_  
(Responsable)

**NOTA :** La relación anexa o diskette deberá contener la información previamente establecida, o en su caso como mínimo la siguiente información en excel.

<u>C O N C E P T O</u>	<u>CREDITO PUENTE</u>	<u>INDIVIDUALIZACION</u>
Número de Asignación	X	X
Nombre del Conjunto Habitacional	X	X
Tasa de Interés	X	
Días Cálculo de Interés	X	
Importe de Intereses en Udi's	X	
Importe de Intereses en Pesos	X	
Importe de Erogaciones Cobradas		X
Importe de Intermediación Cobrada		X
Importe de Intermediación Descontada		X
Otros (Interés Moratorio, etc.)	X	X
Importe Total que nos entregan	X	X

**Figura 3.3.5 Formato del pago mensual de cobranza**

**Solicitud de Crédito Hipotecario**

**DATOS DEL CLIENTE**

Apellido del Cliente  Adquiriente  Concesionario  
 Propio de Puntos  Liquidado  Reconvertido  
Participaciones  Liquidado/Reconvertido  Donador Substituto  
Prestación  Normal  Plus

Nombre completo \_\_\_\_\_  
Carnet del cliente \_\_\_\_\_

**DATOS DESCRITIVOS**

Modelo del inmueble (M) \_\_\_\_\_  
Aparición posesión \_\_\_\_\_  
Aparición inscripción \_\_\_\_\_  
Suelo  S  SPC o Reconvertido  
Clase de edificio \_\_\_\_\_  
E/S-DP \_\_\_\_\_  
No. de unidades de BMS o BMS/2 \_\_\_\_\_  
Edad del bien y parcelas \_\_\_\_\_  
Fecha de inscripción BMS/2 \_\_\_\_\_  
No. de unidades \_\_\_\_\_  
Suelo rural:  Comunal  Individual  Com. Libre  
 Muebler  Industrial

Regimen conyugal:  Sin título Conyugal  Separación de Bienes

Actividad económica:  Comercio  Industria  Servicios de Bienes  
 Negocio  Profesionista  Serviciado  
 Suav. / Participación  Comercio Terrestre  No empleado  
 Programista

Valores de dependencias inmobiliarias

Valor	Dependencia

**DATOS DE LA CUENTA DE BANCOS**

Residencia Compañía \_\_\_\_\_  
Apellido posesión \_\_\_\_\_  
Apellido inscripción \_\_\_\_\_  
BPC o Reconvertido \_\_\_\_\_  
Clase de edificio \_\_\_\_\_  
No. de unidades de BMS o BMS/2 \_\_\_\_\_  
Edad del bien y parcelas \_\_\_\_\_  
Fecha de inscripción BMS/2 \_\_\_\_\_  
No. de unidades \_\_\_\_\_  
Suelo rural:  Comunal  Individual  Com. Libre  
 Muebler  Industrial

Residencia Compañía \_\_\_\_\_  
Apellido posesión \_\_\_\_\_  
Apellido inscripción \_\_\_\_\_  
BPC o Reconvertido \_\_\_\_\_  
Clase de edificio \_\_\_\_\_  
No. de unidades de BMS o BMS/2 \_\_\_\_\_  
Edad del bien y parcelas \_\_\_\_\_  
Fecha de inscripción BMS/2 \_\_\_\_\_  
No. de unidades \_\_\_\_\_  
Suelo rural:  Comunal  Individual  Com. Libre  
 Muebler  Industrial

Residencia Compañía \_\_\_\_\_  
Apellido posesión \_\_\_\_\_  
Apellido inscripción \_\_\_\_\_  
BPC o Reconvertido \_\_\_\_\_  
Clase de edificio \_\_\_\_\_  
No. de unidades de BMS o BMS/2 \_\_\_\_\_  
Edad del bien y parcelas \_\_\_\_\_  
Fecha de inscripción BMS/2 \_\_\_\_\_  
No. de unidades \_\_\_\_\_  
Suelo rural:  Comunal  Individual  Com. Libre  
 Muebler  Industrial

**DATOS DEL INMUEBLE**

Carácter del inmueble  Propio  Compañía  
Código Postal \_\_\_\_\_  
Municipio \_\_\_\_\_  
Cantón \_\_\_\_\_  
Provincia \_\_\_\_\_  
País \_\_\_\_\_

**DATOS DEL PRESTADOR**

Nombre completo \_\_\_\_\_  
Carnet del prestatario \_\_\_\_\_  
Código Postal \_\_\_\_\_  
Municipio \_\_\_\_\_  
Cantón \_\_\_\_\_  
Provincia \_\_\_\_\_  
País \_\_\_\_\_

**DATOS DEL BIEN**

Modelo del inmueble \_\_\_\_\_  
Aparición posesión \_\_\_\_\_  
Aparición inscripción \_\_\_\_\_  
Suelo  S  SPC o Reconvertido  
Clase de edificio \_\_\_\_\_  
E/S-DP \_\_\_\_\_  
No. de unidades de BMS o BMS/2 \_\_\_\_\_  
Edad del bien y parcelas \_\_\_\_\_  
Fecha de inscripción BMS/2 \_\_\_\_\_  
No. de unidades \_\_\_\_\_  
Suelo rural:  Comunal  Individual  Com. Libre  
 Muebler  Industrial

Regimen conyugal:  Sin título Conyugal  Separación de Bienes

Actividad económica:  Comercio  Industria  Servicios de Bienes  
 Negocio  Profesionista  Serviciado  
 Suav. / Participación  Comercio Terrestre  No empleado  
 Programista

Valores de dependencias inmobiliarias

Valor	Dependencia

**DATOS DE LA CUENTA DE BANCOS**

Residencia Compañía \_\_\_\_\_  
Apellido posesión \_\_\_\_\_  
Apellido inscripción \_\_\_\_\_  
BPC o Reconvertido \_\_\_\_\_  
Clase de edificio \_\_\_\_\_  
No. de unidades de BMS o BMS/2 \_\_\_\_\_  
Edad del bien y parcelas \_\_\_\_\_  
Fecha de inscripción BMS/2 \_\_\_\_\_  
No. de unidades \_\_\_\_\_  
Suelo rural:  Comunal  Individual  Com. Libre  
 Muebler  Industrial

Residencia Compañía \_\_\_\_\_  
Apellido posesión \_\_\_\_\_  
Apellido inscripción \_\_\_\_\_  
BPC o Reconvertido \_\_\_\_\_  
Clase de edificio \_\_\_\_\_  
No. de unidades de BMS o BMS/2 \_\_\_\_\_  
Edad del bien y parcelas \_\_\_\_\_  
Fecha de inscripción BMS/2 \_\_\_\_\_  
No. de unidades \_\_\_\_\_  
Suelo rural:  Comunal  Individual  Com. Libre  
 Muebler  Industrial

Residencia Compañía \_\_\_\_\_  
Apellido posesión \_\_\_\_\_  
Apellido inscripción \_\_\_\_\_  
BPC o Reconvertido \_\_\_\_\_  
Clase de edificio \_\_\_\_\_  
No. de unidades de BMS o BMS/2 \_\_\_\_\_  
Edad del bien y parcelas \_\_\_\_\_  
Fecha de inscripción BMS/2 \_\_\_\_\_  
No. de unidades \_\_\_\_\_  
Suelo rural:  Comunal  Individual  Com. Libre  
 Muebler  Industrial

**DATOS DEL INMUEBLE**

Carácter del inmueble  Propio  Compañía  
Código Postal \_\_\_\_\_  
Municipio \_\_\_\_\_  
Cantón \_\_\_\_\_  
Provincia \_\_\_\_\_  
País \_\_\_\_\_

**DATOS DEL PRESTADOR**

Nombre completo \_\_\_\_\_  
Carnet del prestatario \_\_\_\_\_  
Código Postal \_\_\_\_\_  
Municipio \_\_\_\_\_  
Cantón \_\_\_\_\_  
Provincia \_\_\_\_\_  
País \_\_\_\_\_

**DATOS DEL BIEN**

Modelo del inmueble \_\_\_\_\_  
Aparición posesión \_\_\_\_\_  
Aparición inscripción \_\_\_\_\_  
Suelo  S  SPC o Reconvertido  
Clase de edificio \_\_\_\_\_  
E/S-DP \_\_\_\_\_  
No. de unidades de BMS o BMS/2 \_\_\_\_\_  
Edad del bien y parcelas \_\_\_\_\_  
Fecha de inscripción BMS/2 \_\_\_\_\_  
No. de unidades \_\_\_\_\_  
Suelo rural:  Comunal  Individual  Com. Libre  
 Muebler  Industrial

Regimen conyugal:  Sin título Conyugal  Separación de Bienes

Actividad económica:  Comercio  Industria  Servicios de Bienes  
 Negocio  Profesionista  Serviciado  
 Suav. / Participación  Comercio Terrestre  No empleado  
 Programista

Valores de dependencias inmobiliarias

Valor	Dependencia

Figura 3.3.6 Solicitud de un crédito hipotecario

Anexo B

FORMATO DE ESTUDIO SOCIOECONOMICO

[Nombre de la Agencia Investigadora] [Dirección Completa] [Registro Federal de Causantes]		Expediente N°: Investigación: <input type="checkbox"/> Substituto <input type="checkbox"/> Ocasión Substituto Fecha: Mes-Año-Día	
---	--	--	--

<b>Datos del Solicitante</b>			
Nombre (s) completo (s):	Apellido paterno:	Apellido materno:	R. F. C. con conocimiento:
Educación:	Fecha de nacimiento (a-a-a-a):	Nacionalidad (en caso de ser extranjero especificar): <input type="checkbox"/> Mexicana <input type="checkbox"/> Nicaragüense <input type="checkbox"/> Cubana	
Estado civil:	<input type="checkbox"/> Soltero <input type="checkbox"/> Casado <input type="checkbox"/> Unión libre	Algunos momentos: <input type="checkbox"/> Separado <input type="checkbox"/> Viudo <input type="checkbox"/> Casado	
Ocasión de Estudio:	<input type="checkbox"/> Pregrado <input type="checkbox"/> Posgrado <input type="checkbox"/> Especialista <input type="checkbox"/> Substituto	Institución: <input type="checkbox"/> Universidad <input type="checkbox"/> Instituto Tecnológico y de Estudios Superiores de Occidente	
<input type="checkbox"/> Con que Servicio Médico cuenta? <input type="checkbox"/> Ninguno <input type="checkbox"/> IMSS <input type="checkbox"/> ISSSTE <input type="checkbox"/> Otro particular			

<b>Datos del Cónyuge o Concomite (a)</b>			
Nombre (s) completo (s):	Apellido paterno:	Apellido materno:	R. F. C. con conocimiento:
Educación:	Fecha de nacimiento (a-a-a-a):	Nacionalidad (en caso de ser extranjero especificar): <input type="checkbox"/> Mexicana <input type="checkbox"/> Nicaragüense <input type="checkbox"/> Cubana	
Ocasión de estudio:	<input type="checkbox"/> Ninguno <input type="checkbox"/> Pregrado <input type="checkbox"/> Posgrado <input type="checkbox"/> Especialista <input type="checkbox"/> Substituto	Institución: <input type="checkbox"/> Universidad <input type="checkbox"/> Instituto Tecnológico y de Estudios Superiores de Occidente	
<input type="checkbox"/> Con que Servicio Médico cuenta? <input type="checkbox"/> Ninguno <input type="checkbox"/> IMSS <input type="checkbox"/> ISSSTE <input type="checkbox"/> Otro particular			

Fecha de nacimiento, género, o lista de los hijos (en caso de ser libre)

<b>Domicilio del Solicitante</b>			
Calle:	Núm. Exterior:	Nombre o núm. del edificio:	Núm. interior o depto.:
Municipio:	Local:	Ciudad o población:	Delegación o municipio:
Estado:	Código Postal:	Teléfono (s) residencial, y del domicilio y otro para negocios:	
¿Tiempo de vivir en el domicilio actual?	¿Verificación con vivienda de destino siempre tuvo de vivir en el domicilio actual?		
Vive en casa:	<input type="checkbox"/> Propia <input type="checkbox"/> Rentada <input type="checkbox"/> Prestada	Núm. de arrendatarios (en caso de ser inquilino):	Nombre de la institución:
Saldo por pagar:	Pago mensual:	Fecha de hipoteca:	Nombre del arrendador o propietario (en caso de ser inquilino o prestado):
\$	\$	\$	Teléfono (s):

<b>Propietario de arrendador o propietario</b>	
Domicilio completo de arrendador o propietario:	¿Cuánto tiempo ha vivido el solicitante en esa dirección? Años: _____ Meses: _____
¿Paga puntualmente? <input type="checkbox"/> Sí <input type="checkbox"/> No	Comentarios:

<b>Verificación Domicilio</b>			
¿Vive con familiares? <input type="checkbox"/> Sí <input type="checkbox"/> No	¿Vive en buena zona? <input type="checkbox"/> Sí <input type="checkbox"/> No	¿Proporciona algún tipo de ayuda? <input type="checkbox"/> Sí <input type="checkbox"/> No	Monto pagado en buena moneda: \$
Núm. habitaciones de la zona: <input type="checkbox"/> Alto <input type="checkbox"/> Medio <input type="checkbox"/> Bajo	¿Cuentan con servicios de la vivienda? <input type="checkbox"/> Agua <input type="checkbox"/> Gas <input type="checkbox"/> Luz	Tipo de vivienda: <input type="checkbox"/> Departamento <input type="checkbox"/> Casa Solo	<input type="checkbox"/> Casa en arrendamiento
Tamaño aprox. de la vivienda: m <sup>2</sup>	Núm. de espacios habitables:	Núm. de personas que habitan la vivienda:	Servicios relacionados en el interior: <input type="checkbox"/> Agua <input type="checkbox"/> Luz <input type="checkbox"/> Teléfono

<b>Compromisos de vivienda a recibir por el solicitante (Selecciona únicamente el que presenta)</b>			
<input type="checkbox"/> Rendón de luz	<input type="checkbox"/> Rendón de agua	<input type="checkbox"/> Rendón de teléfono	<input type="checkbox"/> Boleto predial
<input type="checkbox"/> Rendón de gas	<input type="checkbox"/> Rendón de pago de hipoteca	<input type="checkbox"/> Otros (especificar):	
Importe de recibos de pago de renta: \$		Importe del recibo pago de hipoteca: \$	

<b>Domicilio Arrendador del Solicitante</b>			
Calle:	Núm. Exterior:	Nombre o núm. del edificio:	Núm. interior o depto.:
Municipio:	Local:	Ciudad o población:	Delegación o municipio:
Estado:	Código Postal:	Teléfono:	
¿Tiempo de vivir en ese domicilio: _____ meses			

Figura 3.3.7 Formato de estudio socioeconómico

<b>BOTAL</b> BANCO INTERNACIONAL			<b>FICHA MULTIPLE DE DEPOSITO</b>		
			<b>MONEDA NACIONAL</b>		
<b>NUMERO DE CUENTA</b>		1234567890	<b>CIUDAD DONDE RADICA SU CUENTA</b>		MEXICO D.F.
<b>FECHA</b>		Mayo 27, 2002	<b>NOMBRE</b>		
<b>CHEQUES A CARGO DE BANCOS LOCALES</b>			<b>CHEQUES A CARGO DE BANCOS FORANEOS</b>		
NÚMERO DE CHEQUE	BANCO EMISOR	IMPORTE	NÚMERO DE CHEQUE	BANCO EMISOR	IMPORTE
1			1		
2			2		
3			3		
4			4		
5			5		
6			6		
7			7		
8			8		
9			9		
10			10		
11			11		
12			12		
13			13		
14			14		
15			15		
16			16		
17			17		
18			18		
19			19		
20			20		
<b>TOTAL 1</b>		<b>\$0.00</b>	<b>TOTAL 2</b>		<b>\$0.00</b>
<b>EFFECTIVO</b>	<b>\$0.00</b>				
<b>TOTAL 1</b>	<b>\$0.00</b>				
<b>TOTAL 2</b>	<b>\$0.00</b>				
<b>TOTAL GENERAL</b>	<b>\$0.00</b>				
<b>NOMBRE DE QUIEN DEPOSITA</b>			ESTE RECIBO SOLO SERA VALIDO CUANDO FIGURE EN EL LA IMPRESION DE NUESTRA MAQUINA REGISTRADORA, SELLO Y FIRMA DEL CAJERO LOS CHEQUES SE RECIBEN SALVO BUEN (MINISTRACIONES) CODIGO		
<b>FIRMA</b>					

Figura 3.3.8 Formato de depósito en el banco

Una vez que la información ha sido recopilada, podemos comprender mejor la situación del sistema de cobranza actual, esto nos facilitará la optimización y la propuesta del nuevo sistema nuevo.

### **Objetivos generales actuales y papel de cada área**

- Dar seguimiento a los pagos por concepto de cobranza con el fin de recuperar los montos esperados en el plazo requerido.
- Realizar el cobro efectivo de los intereses moratorios (cuando es posible) por retraso en los pagos o rechazo de los documentos de cobro.
- Identificar a los Intermediarios Financieros que han incurrido en retrasos para realizar sus pagos (una vez revisado los depósitos en la línea empresarial) y exigir de forma inmediata su liquidación.
- Contar con liquidez de forma oportuna para los fines que a la empresa convengan.

### 3.4. Identificación del problema

Una vez planteada la problemática actual de la empresa, expuestos los requerimientos de ésta para el desarrollo de un sistema que la resuelva y analizado la información disponible, nos ocuparemos de identificar los diversos inconvenientes que se han suscitado por la falta de tal sistema, mismo que pueda manejar toda la información en forma rápida y efectiva.

Por consiguiente, iniciaremos mostrando en la figura 3.4.1 los procedimientos involucrados en los procesos de crédito y cobranza, ya que el segundo tiene su origen en el primero.

#### Crédito y Cobranza

##### Procedimiento de otorgamiento del crédito hipotecario.

Solicitud de crédito hipotecario  
Estudio Socioeconómico  
Otorgamiento del crédito  
Hipotecario

##### Procedimiento de Contratación

Subasta de Recursos  
Asignación de recursos  
Formalización Jurídica del  
otorgamiento de recursos

Notificación de Vivienda  
Adjudicada

Registro en cartera  
**Procedimiento de Cobranza**  
Recopilación de Pagos

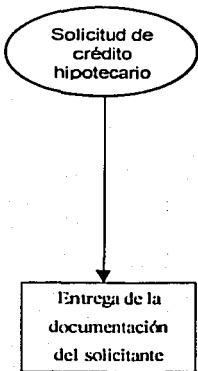
Validación de Pagos

**Procedimiento de verificación  
del cobro**

Deposito de documento de pago  
Seguimiento del Cobro  
Confirmación del pago

**Figura 3.4.1 Procedimientos involucrados en los procesos de crédito y cobranza**

### Procedimiento de Otorgamiento de Crédito



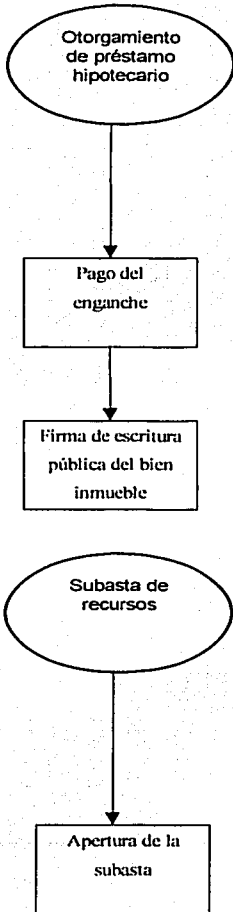
La persona interesada en obtener un crédito hipotecario hace una solicitud del mismo ante un intermediario financiero. Previamente, el solicitante recavó información de los planes de financiamiento existentes, así como de las viviendas en construcción, en caso de ser nueva vivienda, y las características de las mismas.

Cuando el solicitante cuenta con toda la documentación requerida por el intermediario financiero, tal como: comprobante de ingresos, referencias de crédito, éste la entrega para iniciar el trámite. Cabe señalar, que también el solicitante paga un monto por conceptos de gastos de averiguación, determinado por el Intermediario Financiero.



El Intermediario Financiero analiza la solvencia del "solicitante de crédito" para determinar si es una persona sujeta de crédito con base en la documentación aportada y la investigación realizada.

Adicionalmente, el intermediario financiero puede hacer uso de los burós especializados para realizar la investigación crediticia.



Una vez que los resultados de la investigación fueron satisfactorios y el "solicitante de crédito" ha sido considerado una persona sujeta de crédito, se le concede el "apoyo financiero" por la entidad financiera y de crédito, en este caso, a través del intermediario financiero.

Entonces el "acreditado" realiza el pago del enganche, el cual puede ser único o diferido, si la obra no ha sido concluida.

La formalización, ante notario público de la operación de compraventa, tiene como fin sustentar que el apoyo financiero ha sido otorgado.

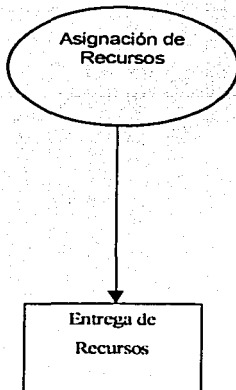
### Procedimiento de Contratación

Los recursos financieros, que posee la empresa hipotecaria, se asignan a través del sistema de subastas a la que están llamados los Intermediarios Financieros registrados y autorizados ante la empresa hipotecaria en cuestión.

La empresa hipotecaria abre la subasta a través de una convocatoria y pide a los Intermediarios Financieros cumplan con los siguientes requisitos para poder participar:

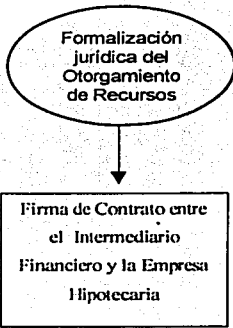


- Registro del Intermediario Financiero actualizado de la empresa hipotecaria.
- Registro del proyecto.
- La documentación señalada en la convocatoria de subasta correspondiente.

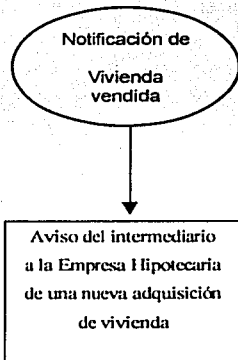


El Intermediario financiero ganador de la subasta recibe la asignación de derechos sobre los créditos de largo plazo para los futuros adquirientes de las viviendas que edificarán.

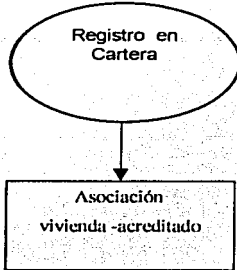
Los derechos sobre los créditos derivados de las subastas, sólo podrán ser ejercidos por Intermediarios Financieros que hayan recibido asignación de recursos, para canalizarlos a los acreditados finales, una vez que estos hayan cumplido con los requisitos establecidos por la empresa hipotecaria y los Intermediarios Financieros.



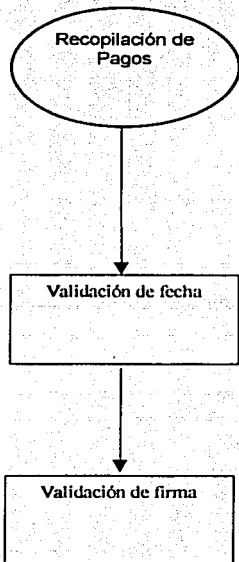
La formalización de la asignación de los recursos se realiza mediante la firma de un contrato entre el Intermediario Financiero ganador de la subasta y la empresa hipotecaria, en el cual se establecen las condiciones generales del Financiamiento y garantías a las instituciones de Crédito.



Después, se realiza la entrega formal de los recursos para el otorgamiento del préstamo hipotecario al "acreditado" por conducto del Intermediario Financiero. Éste avisará a la empresa hipotecaria que una vivienda ha sido vendida toda vez que los requisitos establecidos por ambos han sido satisfechos, para proceder al registro en cartera.



La Empresa hipotecaria realiza el registro en la cartera del nuevo acreditado, especificando el número del contrato, las características de la vivienda y las condiciones del crédito, tales como, monto, plazo y tasa. Es decir, se genera la información necesaria para continuar posteriormente con el proceso de cobranza.



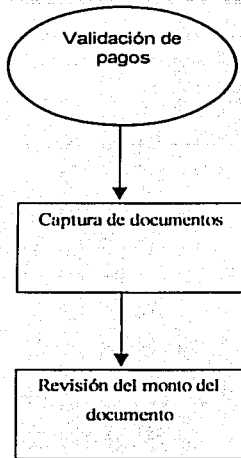
### Procedimiento de Cobranza

El personal del departamento de cobranza recibe durante el día los cheques, por diversos conceptos acompañados de un documento en el que se indica el concepto del pago, el número del cheque, institución bancaria, Intermediario Financiero que realiza el pago, el monto y otros.

Primeramente, se valida de forma visual, que la fecha no exceda la fecha de recepción del documento.

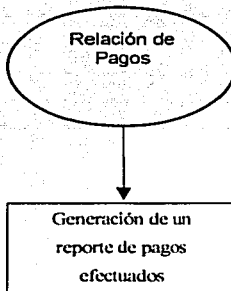
A su vez, se verifica que la firma que aparece en el cheque se encuentre registrada en el catálogo de la empresa.

Cabe señalar que los cheques son aceptados sin importar el monto.

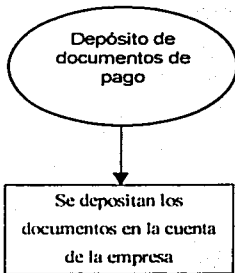


La validación de los pagos recibidos se efectúa con la relación de documentos, capturados en una hoja de cálculo, contra un reporte que indica la cantidad que debió ser cubierta por el Intermediario Financiero. Es importante señalar, que la captura de la información esta sujeta a errores humanos.

Bajo este método, el usuario revisa que el monto indicado en el cheque sea el esperado, en su defecto si la cantidad es mayor se devuelve el excedente y se notifica al Intermediario Financiero; de lo contrario, se le solicita al Intermediario Financiero vía telefónica envíe la diferencia lo antes posible.

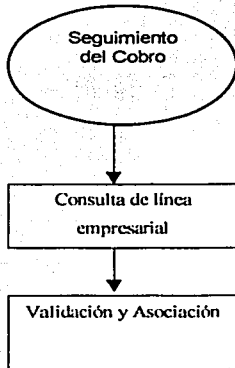


Una vez terminado el procedimiento de validación, el usuario elabora una relación de los pagos efectuados, en el que se incluyen las diferencias entre los montos recibidos y esperados. Este proceso es un poco lento y engorroso porque el cliente lo realiza casi en forma manual y puede cometer errores durante el proceso. Por último, el usuario pasa esta relación al departamento de Finanzas anexando los documentos de pago.



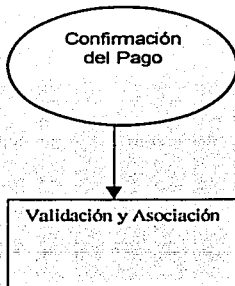
### Procedimiento de Aseguramiento del cobro

Considerando lo anterior, el departamento de Finanzas deposita los documentos de pago en la cuenta bancaria de la empresa.



Con respecto al seguimiento de cobro, el usuario del departamento de Finanzas realiza una consulta a través de una línea empresarial para verificar todos los movimientos bancarios registrados en la cuenta de la empresa, y así generar un archivo donde queden registrados tales movimientos.

El siguiente paso es validar y asociar los pagos recibidos en el archivo de la hoja de cálculo, contra el estado de cuenta electrónico.



Finalmente, el departamento de Finanzas se encarga de verificar que los documentos no hayan sido rechazados ya que de ser así procederá a hacer el cálculo de los intereses moratorios.

Como se podrá apreciar el problema en el proceso de cobranza radica en que toda la información se maneja en hojas de cálculo, donde las búsquedas de información son muy lentas y tediosas pues los registros almacenados son demasiados; además, la existencia de errores en la captura de la información ocasiona que los reportes sean erróneos y por tanto se elaboren nuevamente, con lo cual se pierde tiempo y dinero.

### 3.5 Entidades involucradas en el proceso

Las entidades involucradas en el proceso de cobranza se muestran en la figura 3.5.1.

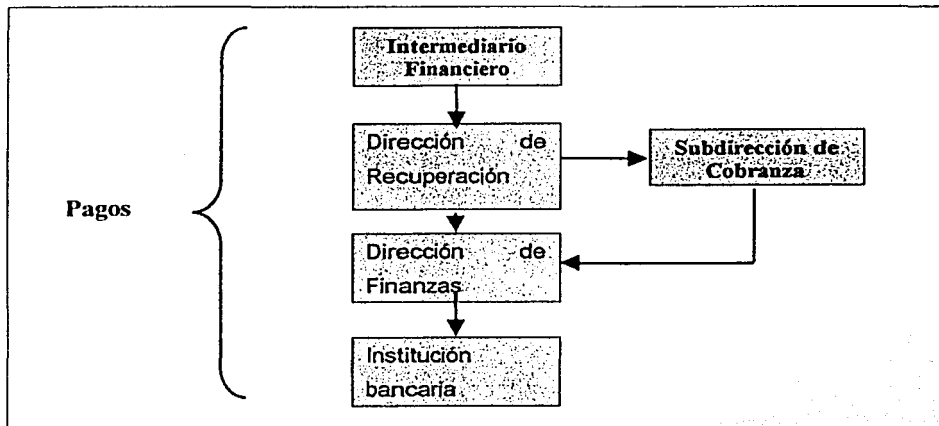


Figura 3.5.1 Entidades involucradas en el proceso del pago

Cada una de estas entidades tiene una función específica, que es necesario conocer como planteamiento de la situación actual:

#### 3.5.1 Intermediario Financiero

La tarea del Intermediario financiero comienza con la etapa de la generación de crédito, mediante el cual, junto con una constructora, poseen un conjunto de viviendas, terminadas o no, listas para subastar.

El intermediario financiero participa en las subastas abiertas por una empresa hipotecaria con el fin de poner a disposición sus viviendas y recuperar la inversión así

como obtener ganancias. En caso de que la empresa hipotecaria opte por un Intermediario financiero, tendrán que formalizar un contrato mediante jurídico con el fin de llevar un mejor control tanto de pagos como de clientes.

El siguiente paso corresponde al Intermediario financiero, el cual tiene que generar informes de ventas a diversos clientes, establecer solicitudes de crédito, estudios socioeconómicos y establecer un contrato con el cliente, identificando la vivienda adquirida. Dicho contrato debe especificar crédito, comisión a cobrar, tasa, plazo, monto de la mensualidad y monto total del crédito.

El intermediario financiero, envía a la empresa hipotecaria una notificación de que una vivienda fue vendida, la empresa hipotecaria registra dicho contrato en cartera, para comenzar el proceso de cobranza a partir de la primera mensualidad.

El intermediario financiero envía diariamente cheques acompañados de un documento, en el cual se indica el concepto de pago, el número de cheque y el monto que espera recibir en la fecha estipulada en el contrato; lo anterior debe estar debidamente firmado.

### **3.5.2 Subdirección de Cobranza**

La Subdirección de cobranza corresponde al área de recuperación, pero nos enfocaremos a esta subdirección, debido a que nos interesa seguir el flujo de la información que proviene del intermediario financiero.

El proceso de cobranza comienza hasta la primera mensualidad y termina hasta que el monto total del crédito es cubierto. No entraremos en detalle a definir el proceso siguiente, ya que se ha especificado en otros capítulos, pero si mencionaremos algunas de las tareas específicas de esta área con respecto a los pagos realizados por el cliente:



- Supervisar la correcta aplicación de intereses de acuerdo a las Condiciones Generales.
- Supervisar la recuperación oportuna de las erogaciones mensuales, pago de intereses y amortización de los créditos.
- Dar seguimiento a la recuperación de los créditos no liquidados o con atraso en el pago.
- Supervisar el cobro de intereses moratorios en los casos en que los intermediarios no entreguen a la Empresa en los tiempos establecidos, los recursos de los créditos.
- Controlar el pago de intermediación financiera y el cobro de comisiones por intermediación en los créditos otorgados.

### **3.5.3 Dirección de Finanzas**

La Dirección de Finanzas, recibe de la subdirección de Cobranza una relación que incluye el número de cheque, institución bancaria y monto. Con esta información, el área de Finanzas genera las fichas para enviarlas a un depósito bancario.

El proceso que sigue ya fue descrito en otro capítulo, sin embargo mencionaremos algunas de las tareas específicas de esta área con respecto a los pagos realizados por el cliente:

- Monitorea la evolución del Entorno de la empresa, compuesto por los mercados de hipotecas y capitales.
- Participa en la definición y evaluación de las fuentes de financiamiento.
- Analiza el riesgo de tasas de mercado.
- Analiza los riesgos de crédito.
- Participa en la programación, seguimiento y evaluación de la gestión financiera de la Empresa y el seguimiento de la cobranza a los intermediarios financieros.

### **3.5.4 Institución Bancaria**

La Institución bancaria es la última parte del proceso del pago y sólo se involucra en el depósito y en el abono de la siguiente forma:

Los cheques que se emiten por la misma institución bancaria donde se depositan, son abonados al momento del depósito, si se comprobó ya la disposición de fondos, pero los que pertenecen a otra institución son abonados salvo buen cobro y es hasta el día siguiente cuando se puede disponer del monto especificado.

### **3.6 Opciones de solución y elección de la óptima**

Durante la búsqueda de posibles soluciones para agilizar el proceso de cobranza y analizando la situación actual, nuestras opciones de solución fueron muy diversas y se comentarán algunas de ellas, sin embargo, los criterios para la elección de la propuesta óptima fueron establecidos en base al equipo con el que la empresa cuenta actualmente, así como las licencias con las que hace actualmente sus procesos.

Las actividades propuestas que responden a los requerimientos de la empresa y que se adaptan a la infraestructura actual fueron las siguientes:

- Captura de datos al recibir los documentos de pago.
- Carga de movimientos bancarios recibidos en la cuenta de la empresa, a través de un archivo que se genera a partir de una línea empresarial.
- Validación y asociación de los pagos recibidos contra el estado de cuenta.
- Reporte de diferencias entre montos recibidos y esperados.
- Reporte de:
  - Estado de los pagos
  - Cheques cobrados
  - Cheques rechazados
  - Pendientes de cobro
- Reporte de monto por concepto de intereses moratorios, (en caso de cheques rechazados y pagos parciales) mediante el intermediario financiero.

- Mantenimiento a catálogos del sistema.
- Control de seguridad

A continuación se explicará cada uno de los puntos anteriores con el fin de explicar de manera más clara el proceso de cobranza automático propuesto:

### **3.6.1 Captura de datos**

El proceso de captura de datos tiene como finalidad reducir el tiempo en los procesos posteriores, como por ejemplo, reportes, consultas y algunas otras tareas que pueden retrasar los tiempos. Sugerimos, en un inicio, la captura mediante algún código, o incluso, mediante escáner, pero consideramos que la empresa y el sistema actual está lejos de contar con tal avance.

La captura de datos, se hará mediante un formato visual sencillo para el usuario, y contará con un catálogo, el cual, dentro de lo posible, validará que la información introducida por el usuario sea la correcta.

La pantalla para el ingreso de datos pedirá la información mínima necesaria, como por ejemplo:

- Número de cheque
- Institución emisora (intermediario financiero)
- Fecha del documento
- Fecha de registro
- Tipo de pago
- Concepto de pago
- Monto del pago

- Mantenimiento a catálogos del sistema.
- Control de seguridad

A continuación se explicará cada uno de los puntos anteriores con el fin de explicar de manera más clara el proceso de cobranza automático propuesto:

### **3.6.1 Captura de datos**

El proceso de captura de datos tiene como finalidad reducir el tiempo en los procesos posteriores, como por ejemplo, reportes, consultas y algunas otras tareas que pueden retrasar los tiempos. Sugerimos, en un inicio, la captura mediante algún código, o incluso, mediante escáner, pero consideramos que la empresa y el sistema actual está lejos de contar con tal avance.

La captura de datos, se hará mediante un formato visual sencillo para el usuario, y contará con un catálogo, el cual, dentro de lo posible, validará que la información introducida por el usuario sea la correcta.

La pantalla para el ingreso de datos pedirá la información mínima necesaria, como por ejemplo:

- Número de cheque
- Institución emisora (intermediario financiero)
- Fecha del documento
- Fecha de registro
- Tipo de pago
- Concepto de pago
- Monto del pago

- Usuario de captura

La pantalla descrita anteriormente quedaría como se indica en la siguiente figura, haciendo la aclaración que no es la pantalla ya definida en el nuevo sistema, pero que si nos da una idea más clara de lo que hemos propuesto a la empresa hipotecaria.

Más adelante se explicará el producto a utilizar para la elaboración de pantallas, así como el manejador de bases de datos a utilizar y el motivo de la elección, ajustándonos también a los requerimientos de la empresa y a las licencias con las que cuenta, ya que esto facilitará la elaboración del nuevo sistema y se ajustará a las necesidades del cliente.

The screenshot shows a window titled 'Empresa Hipotecaria' with a subtitle 'Area de Cobranza'. Below this, it says 'Pantalla de captura de pagos'. The form contains several input fields: 'Número de Cheque' (empty), 'Clave de Institución Emisora' (with a dropdown menu showing 'Intermediario Financiero'), 'Fecha del documento' (with a date field containing '01/01/2002'), 'Fecha de registro' (with a date field containing '01/01/2002'), 'Tipo de pago' (with a dropdown menu), 'Concepto de pago' (with a dropdown menu), 'Monto' (with a field containing '\$'), 'Clave de Usuario de captura' (empty), and 'Estatus' (empty). At the bottom, there are two buttons: 'Aceptar' and 'Reestablecer'.

Figura 3.6.1.1 Pantalla para la captura de datos

Los datos validados contra catálogo serán la clave del intermediario financiero, el tipo de pago, el concepto de pago, el usuario y el estatus.

### **3.6.2 Carga de movimientos**

La carga de este archivo en los movimientos de la empresa, agilizará el proceso de cobranza y nos ayudará tanto en la validación de datos como en la generación de reportes, sobre la línea empresarial, el archivo será cargado a través de una opción en el sistema donde importaremos el archivo texto generado por la terminal de la línea empresarial, donde será cargado en un lapso estimado de un día.

La línea empresarial es un enlace dedicado, es decir, cada vez que se desee consultar ya sea a un cliente, un monto o cualquier tipo de dato, la información estará disponible en todo momento.

### **3.6.3 Validación y asociación de los pagos recibidos contra el estado de cuenta**

Dado que, según la estructura de la base de datos, los primeros campos corresponden a la llave primaria, se tienen que validar entre otras cosas los formatos de las fechas (por ejemplo: dd/mm/yyyy), el monto capturado (que sean sólo números).

Debido a que se recibe un documento enviado por el intermediario financiero con los datos generales como el número de cheque, concepto de pago, monto, institución bancaria, intermediario financiero que realiza el pago, clave del depósito bancario.

Se asocian los datos esperados contra los datos capturados en el día y se hace una validación, con el propósito de determinar cuáles pagos han sido cubiertos y cuáles están pendientes de cobro.

### **3.6.4 Reportes**

Por lo anterior, es necesario obtener un reporte que incluya los montos recibidos durante el día contra los montos esperados.

El reporte nos indicará si el pago fue mayor, menor, si fue correcto, o bien, si no se realizó.

Se obtendrá también un reporte de todos aquellos pagos que fueron mayores al esperado, para dar aviso al intermediario, vía telefónica, que le será devuelta la diferencia, dentro de los próximos tres días, mediante un depósito directo en su cuenta bancaria.

El siguiente reporte es el que nos indicará cuáles pagos fueron menores al esperado, ya que en este caso el proceso es diferente al anterior. La diferencia se deposita en cuentas de deudor y se exige el pago inmediato, vía telefónica, y de ser posible, se cobran intereses moratorios.

El archivo nos puede ayudar también a obtener diferentes tipos de reportes como los siguientes:

- Estado de los pagos
- Cheques cobrados
- Cheques rechazados
- Pendientes de cobro
- Reporte de monto por concepto de intereses moratorios

### **3.6.5 Mantenimiento a catálogos del sistema**

Un catálogo es el registro de la información que es común dentro de varias aplicaciones y que no cambian, en una tabla. Son utilizados para facilitar la captura de datos al usuario, con dos fines: para que tenga que teclear menos y para evitar los errores humanos, ya que pueden equivocarse por "errores de dedo". El mantenimiento de los catálogos es una opción de Altas, Bajas y Cambios, ya que se debe mantener actualizado.

### **3.6.6 Seguridad**

La seguridad estará dada de la siguiente forma: Los jefes de cada área tendrán acceso total al sistema y serán ellos quienes determinen el nivel de acceso de cada usuario.

### **3.6.7 Evaluación de productos para el desarrollo del Back – End**

Para facilitar la toma de decisión realizamos una evaluación de los productos más representativos en el mercado, haciendo la diferencia entre los productos para la administración de bases de datos y los productos para llevar a cabo la interfaz. Para poder contar con un manejador de bases de datos relacionales que administre la información de manera eficiente, hicimos una evaluación de los productos más significativos en el mercado: SQL de ORACLE 9i, MICROSOFT SQL SERVER 2000 y MySQL 4.0.1 de SYBASE.

#### **3.6.7.1 Descripción de SQL de Oracle 9i**

Es un producto enfocado a soportar construcciones robustas, basado en un esquema cliente/servidor, cuenta con queries distribuidos y es compatible con plataformas Unix, Windows NT o 2000 Server, VAX, VMS, Windows 95 OS/2 y Macintosh.

El sistema de seguridad es independiente del sistema operativo, y tiene, por lo menos, un nivel C2, cuenta con mecanismos de Auditoría a nivel de usuarios y de objetos de la base de datos, soporta grants y roles. El RDBMS garantiza la recuperación de la información en caso de falla del equipo, el sistema puede respaldar en línea y es posible recuperar la información de un respaldo sin necesidad de dar de baja la base de datos.

Dispone de un ambiente gráfico para la administración que incluye el manejo para bases de datos distribuidas, permite hacer una instalación remota y tiene soporte del estándar SNMP.



Es fácil establecer comunicación con aplicaciones que utilicen conectividad ODBC (Open Database Connectivity).

Garantiza la integridad de los datos, llevando un control de los registros que han sido modificados dentro de una transacción a fin de poder deshacer los cambios en caso de que por algún motivo el proceso no se termine completamente.

Se soporta una gran cantidad de usuarios concurrentes, ejecutando una variedad de aplicaciones de bases de datos. Puede controlar selectivamente la disponibilidad de los datos, según el nivel y subnivel de la base de datos.

Posee las herramientas para desarrollar en WEB .

#### **3.6.7.2 Descripción de SQL Server 2000**

SQL Server 2000 contiene soporte de XML y HTTP que simplifica el acceso a los datos y el intercambio de los mismos, además de tener una mayor capacidad de análisis de los datos. Lo cual maximiza el tiempo de funcionamiento y las funciones avanzadas de administración automatizan las tareas rutinarias. A través de las herramientas de programación y servicios se aceleran los tiempos de desarrollo.

La mayor ventaja es la fuerte integración con la plataforma .NET, para integrar y desarrollar cualquier negocio en Internet/Intranet.

Tiene una mayor simplificación de la integración de sistemas de back-end y transferencia de datos a través de firewalls con SQL Server 2000 XML y otros estándares de Internet. Se puede acceder a los datos utilizando XML sin utilizar programación de bases de datos relacionales, mientras que los administradores de bases de datos pueden manipular fácilmente datos, también en formato XML, utilizando Transact-SQL (T-SQL) y procedimientos almacenados.

Facilita la conexión a bases de datos SQL Server 2000 y cubos OLAP, flexiblemente utilizando la Web sin programación adicional. Utiliza la conectividad de datos con HTTP para garantizar la consulta de los datos, incluso por usuarios finales, a través de una URL e interfaces de usuario intuitivos.

Copia y mueve bases de datos entre servidores o instancias sin tiempo de caída a través del nuevo Asistente para copia de Base de Datos. Contiene un Analizador de Consultas SQL y Servicios de Transformación de Datos que amplían la capacidad de mover y transformar datos desde cualquier fuente.

Aumento de la productividad con las mejoras de T-SQL. Las nuevas funciones definidas por el usuario para reutilizar código y simplificar el desarrollo. Mantenimiento de datos consistentes con integridad referencial en cascada para controlar la propagación de operaciones de borrado y actualización.<sup>1</sup>

### 3.6.7.3 Descripción de MySQL 4.0.1 de Sybase

MySQL es el sistema de base de datos utilizado por Sybase, tiene el mejor desempeño y escalabilidad, comparado con los diferentes manejadores de bases de datos, como Oracle, SQL Server y DB2, por ejemplo, hablando de velocidad, es capaz de ejecutar 500 páginas por segundo, a diferencia de Oracle con 400 páginas por segundo, SQL Server 2000 también con 400 páginas por segundo y finalmente 200 páginas por segundo de DB2.

Los drivers de MySQL tienen una excelente combinación, ya que utiliza JDBC (Java Database Connectivity) que es completo, fijo y estable.

MySQL 4.0.1 garantiza la integridad de datos y contiene auxiliares que ayudan a la configuración de las bases de datos así como a la configuración cliente/servidor, tal es el caso del archivo my.cnf que auxilia en esta tarea.

---

<sup>1</sup> [www.microsoft.com/sql](http://www.microsoft.com/sql)

El aspecto más importante de MySQL es el mecanismo por el cual asegura los datos y la integridad. Cuenta con privilegios muy específicos tanto para el administrador de la base de datos como para los usuarios.

Contiene un programa especial para el cliente, llamado MySQL Monitor, es una interface que permite al cliente conectarse a la base de datos, ejecutar queries y ver sus resultados. Garantiza, además la seguridad en password.

Contiene herramientas como *mysqladmin* que es usado para administrar varios aspectos del servidor de base de datos MySQL, como crear y borrar tablas.

Tiene, además herramientas para hacer backups (respaldos) de las bases de datos, como el *mysqldump* y el *mysqlhotcopy*, las cuales facilitan la estructura de bases de datos.<sup>2</sup>

### 3.6.7.4 Comparación de productos Back-End.

Tomamos en consideración los DBMS descritos anteriormente, presentamos un cuadro descriptivo del análisis de los productos:

	ORACLE 9i	SOL SERVER 2000	MySQL
<b>Modelos de Control de Datos</b>			
Opciones de violación de integridad referencial	Restriictiva, excepto en el borrado en cascada	Sólo restrictiva	Restriictiva en los privilegios para las tablas
Vistas actualizables con opción de verificación	SI	SI	SI
Clausula where referencial	NO	NO	NO
<b>Opciones de Control de Datos</b>			
Tipos de datos definidos por el usuario	SI	SI	SI
BLOB's	SI	SI	NO
Binary large objects			

<sup>2</sup> W.J. Gilmore, A Programmer's Introduction to PHP 4.0

El aspecto más importante de MySQL es el mecanismo por el cual asegura los datos y la integridad. Cuenta con privilegios muy específicos tanto para el administrador de la base de datos como para los usuarios.

Contiene un programa especial para el cliente, llamado MySQL Monitor, es una interface que permite al cliente conectarse a la base de datos, ejecutar queries y ver sus resultados. Garantiza, además la seguridad en password.

Contiene herramientas como mysqladmin que es usado para administrar varios aspectos del servidor de base de datos MySQL, como crear y borrar tablas.

Tiene, además herramientas para hacer backups (respaldos) de las bases de datos, como el *mysqldump* y el *mysqlhotcopy*, las cuales facilitan la estructura de bases de datos.<sup>2</sup>

#### 3.6.7.4 Comparación de productos Back-End.

Tomamos en consideración los DBMS descritos anteriormente, presentamos un cuadro descriptivo del análisis de los productos:

	ORACLE 9i	SQL SERVER 2000	MySQL
<b>Modelo de Integridad</b>			
Opciones de violación de integridad referencial	Restriictiva, excepto en el borrado en cascada	Sólo restrictiva	Restriictiva en los privilegios para las tablas
Vistas actualizables con opción de verificación	SI	SI	SI
Cláusula where referencial	NO	NO	NO
<b>Objetos de Base de Datos</b>			
Tipos de datos definidos por el usuario	SI	SI	SI
BLOB's	SI	SI	NO
Binary large objects			

<sup>2</sup> W.J. Gilmore, *A Programmer's Introduction to PHP 4.0*

<b>Tipos de datos soportados</b>	Imagen, video, texto, messaging, spatial, data types	Imagen, video, texto, money, bit, binary	Imagen, video, texto, money, bit, binary
<b>Estructura de tablas</b>	Heap y clustered	Sin elección	Mediante <i>mysqldump</i>
<b>Facilidad de optimización</b>	Distribución de índices de tablas	Fill factors	Mediante my.cnf y <i>mysqldump</i>
<b>Tipos de bases de datos</b>			
<b>Nivel</b>	Row based (Rules), set based	Set based	Set based
<b>Anidados</b>	SI	SI	SI
<b>Procedimiento almacenado</b>			
<b>Lenguaje</b>	PL/SQL	TRANSSACT SQL	MySQL Database Server
<b>Anidados</b>	SI	SI	SI
<b>Sintaxis ANSI en Outer Joins</b>	NO	SI	SI
<b>Administración de bases de datos</b>			
<b>Herramientas</b>	Oracle Enterprise Manager, Performance pack	Enterprise Manager Performance Monitor Analizador de consultas <u>Data mining</u>	PHP
<b>Soporta SNMP (Simple Network Management protocol)</b>	SI	SI	SI
<b>Soporte a Internet</b>			
<b>Herramienta de soporte a Internet</b>	Oracle web Server	Internet Information Server en Windows NT	PHP3
<b>A otros DBMS</b>	A través de gateways	A través de ODBC	A través de MyODBC
<b>Costo</b>			
<b>Velocidad</b>	BUENA	MUY BUENA	MUY BUENA
<b>Costo</b>	REGULAR	BAJO	BAJO

Figura 3.6.7.4.1 Tabla comparativa entre diferentes DBMS <sup>3</sup>

<sup>3</sup> [www.mysql.com](http://www.mysql.com), [www.otri.us.es/recursos/PHP/manual/odbc/importar\\_tablas.htm](http://www.otri.us.es/recursos/PHP/manual/odbc/importar_tablas.htm)

	ORACLE	MySQL	SQL SERVER
<i>Facilidad para utilizar la herramienta</i>	regular	bueno	bueno
<i>Rapidez</i>	bueno	bueno	bueno
<i>Disponibilidad de la herramienta y costo</i>	regular	regular	bueno
<i>Integración con la plataforma de desarrollo de Windows</i>	bueno	bueno	bueno
<i>Compatibilidad con otros lenguajes de desarrollo</i>	bueno	bueno	bueno

Figura 3.6.7.4.2 Cuadro comparativo del back-end.

### 3.6.8 Evaluación de productos para el desarrollo del Front-End

Para la construcción del FRONT-END se requiere la construcción de un sistema capaz de utilizar la base de datos de una manera eficiente y a la vez con un formato sencillo para el usuario final, que pueda presentar los resultados de una manera amigable. Para lograr tal objetivo hemos optado por el producto de PowerBuilder 7.0.1 de Sybase Powersoft; sin embargo, consideramos también los productos de Visual Basic 6.0 y Delphi 5 de Borland. A continuación describiremos cada uno de estos productos de manera breve y presentaremos una tabla de las ventajas de PowerBuilder sobre los demás.

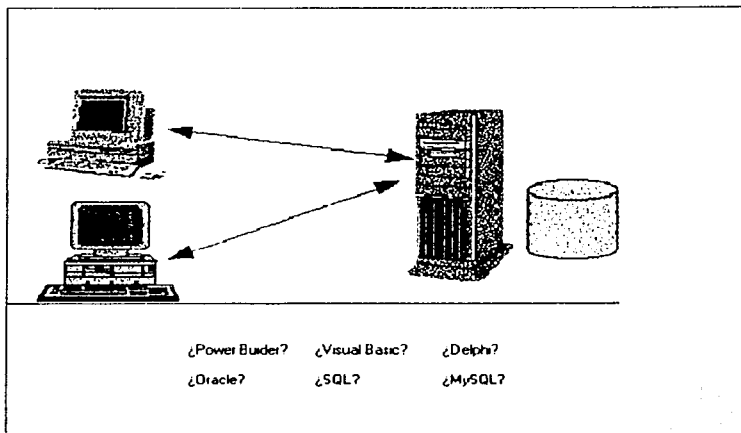


Figura 3.6.8.1 Toma de decisión sobre el producto para Front-End

### 3.6.8.1 PowerBuilder

PowerBuilder fue uno de los primeros ambientes de programación de aplicaciones Cliente/Servidor en plataformas gráficas en salir al mercado y quizás debido a ello, es una de las más difundidas. La versión actual de PowerBuilder es la 7.0.1. La programación en PowerBuilder se hace a través de **painters** (graficadores) de diferentes tipos de objetos: Ventanas, Menús, Funciones y Aplicaciones.

Para la elaboración de bases de datos, PowerBuilder introduce un objeto llamado *Datawindow*. Un *datawindow* es un objeto que encapsula funciones de presentación de los datos (interfaz usuario) y de interacción con bases de datos. Especificando una sentencia SQL, el *datawindow* evita una gran labor de programación al proveer funcionalidades automáticas sobre los datos (inserción, modificación, borrado) e introduce un gran conjunto de instrucciones y eventos para la manipulación de los

datos en un datawindow. Hay una limitación importante de los datawindows y es que sólo pueden tener asociada una única sentencia SQL, lo que puede ser muy restrictivo en algunos casos.

El painter de datawindows que provee PowerBuilder permite definir, de manera muy amigable, sentencias SQL haciendo uso de una interfaz gráfica con la que es posible definir las diferentes sentencias sin necesidad de conocer en detalle la base de datos (más aún, sin necesidad de dominar SQL para consultas sencillas). Cada datawindow se conecta a la base de datos a través de objetos llamados *Transaction Objects*, lo que le permite al programador controlar el número de conexiones que tendrá cada cliente con el servidor (o servidores) así como también definir parámetros de conexión especiales. El manejo transaccional se hace a través de primitivas *commit* y *rollback* que están disponibles sobre los *Transaction Objects*.

Para el manejo de la interfaz usuario, PowerBuilder permite definir y usar muy fácilmente objetos de la interfaz Windows. Adicionalmente, permite crear objetos nuevos a partir de objetos preexistentes que luego pueden utilizarse en diferentes aplicaciones (User Objects). Igualmente permite la utilización de controles VBX y OCX. Sin embargo, no hay tanta flexibilidad para elaboración de pantallas como la existente en otras herramientas del mundo Windows, como Visual Basic o Delphi.

Para una programación básica, el entrenamiento requerido para conocer PowerBuilder es muy corto. Sin embargo, para una programación avanzada que permita aprovechar realmente la herramienta es necesario un tiempo de aprendizaje un poco mayor.

Las aplicaciones generadas por PowerBuilder, requieren ser interpretadas. Por esta razón es necesario distribuir además de los ejecutables, el ambiente para cliente en producción (Deployment Kit) que es de distribución gratuita. Se puede generar código compilado de partes de la aplicación (expresiones matemáticas, procesamiento de arreglos y llamado a funciones, pero seguirán habiendo partes que requieren ser interpretadas.



PowerBuilder existe en tres versiones: Desktop, ODBC y Enterprise. La versión desktop, como su nombre lo indica, es una versión de escritorio que no permite conectividad con bases de datos externas al computador de desarrollo. La versión ODBC permite el desarrollo de aplicaciones Cliente/Servidor pero restringe la conectividad para que sólo pueda hacerse vía ODBC. La versión Enterprise, permite la conectividad con ODBC o a través de drivers propietarios que permiten obtener un mejor desempeño de la aplicación.<sup>4</sup>

### 3.6.8.2 Delphi

Existen dos tipos de programación sobre Delphi: Programación RAD y Programación Avanzada. La programación RAD de Delphi, recoge el concepto de programación por componentes de Visual Basic. Los componentes son objetos que se constituyen en los bloques de construcción de aplicaciones en Delphi y representan partes visibles de la aplicación (objetos de interfaz como botones, barras de avance, entre otros) y partes no visibles (bases de datos, timers, entre otros).

Para el acceso a bases de datos, Delphi provee varios tipos de componentes con diferentes funcionalidades (*Data Access*): interacción con tablas, definición de sentencias SQL, stored procedures, componentes de Windows pero que tienen la funcionalidad adicional de interactuar contra una base de datos (*Data Controls*).

El programador puede definir el tipo de manejo transaccional que desee o definir diferentes tipos de parámetros para la conexión a la base de datos.

Delphi provee expertos y galerías, que permiten definir formas completas con funcionalidades sofisticadas (como consultas maestro detalle). Con Delphi es muy fácil alcanzar un nivel de productividad apropiado cuando se maneja al nivel de Programación RAD.

---

<sup>4</sup> [agameoon.uniaandes.edu.co/sistemas/6601.htm](http://agameoon.uniaandes.edu.co/sistemas/6601.htm)

Una característica muy interesante de Delphi es que es posible aumentar la funcionalidad de un componente o crear componentes nuevos usando mecanismos de herencia. Este es el nivel de programación avanzada. Con este mecanismo, un programador experimentado puede crear nuevos componentes dentro del mismo ambiente. El programador avanzado puede también crear expertos que faciliten determinadas labores al programador RAD.

Al igual que la mayoría de los ambientes, Delphi permite la utilización de controles VBX (u OCX a partir de Delphi 2.0). Sin embargo, a diferencia de los otros ambientes, la integración es completamente transparente para el programador, hasta el punto de que éste puede no saber si está trabajando con un VBX o un control nativo de Delphi.

Otras características de Delphi que vale la pena mencionar son: permite hacer manejo de errores mediante la utilización de un mecanismo de manejo de excepciones bastante claro, los aplicativos generados son 100% ejecutables por lo que no se requiere un kernel de interpretación al distribuir la aplicación, permite generar DLL estándar de Windows lo que permite disminuir el tamaño y requerimientos de memoria del aplicativo, ofrece mecanismos avanzados de depuración (no sólo depuración del programa que se está construyendo, sino también a nivel Windows mediante un mecanismo de seguimiento de los eventos que se producen).

En cuanto a reportes, Delphi incluye ReportSmith, que permite definir múltiples reportes los cuales pueden ser invocados desde la aplicación por medio de un componente diseñado para ello. (Los reportes, a diferencia de la aplicación son interpretados).

A diferencia de los anteriores ambientes, Delphi sólo está disponible únicamente para plataformas de la familia Windows.<sup>5</sup>

---

<sup>5</sup> [agamenon.uniandes.edu.co/sistemas/6601.htm](http://agamenon.uniandes.edu.co/sistemas/6601.htm)

### 3.6.8.3 Visual Basic 6.0

Es un lenguaje de programación de entorno visual que ha sido muy aceptado por su flexibilidad y velocidad. Es un lenguaje interactivo que permite que su aprendizaje sea rápido.

Utiliza varias formas para sus representaciones visuales de sistemas, una "forma" en Visual Basic es una ventana asociada a controles, iconos, gráficos y código.

Visual Basic 6.0 soporta la abstracción, encapsulación, polimorfismo, y la reutilización de código. Los objetos en Visual Basic están encapsulados, es decir, tienen su propio código.

Además de la interfaz predeterminada, los objetos pueden implementar interfaces adicionales para proporcionar polimorfismo.

La interacción con bases de datos es fácil con Visdata, que permite crear y modificar bases de datos y permite crear aplicaciones para visualizar, editar y actualizar información de varios tipos de bases de datos.

Existen tres ediciones de Visual Basic: la estándar, que incluye controles y cuadro de diálogo estándar. La profesional, que incluye controles activos (*Active X*) adicionales, controles para internet y Crystal Reports. La edición empresarial contiene la galería de objetos y herramientas de administración de bases de datos.

Las herramientas que contiene para facilitar el diseño son: barra de menús, menús contextuales, barra de herramientas, explorador de proyectos, ventana de propiedades, caja de herramientas, formulario, ventana de posición del formulario y examinador de objetos.

Visual Basic permite configurar la interfaz para un solo documento (SDI) o interfaz de múltiples documentos (MDI). Tiene también el acoplamiento entre ventanas. También incluye documentación que ayuda durante la aplicación, son tres los tipos de ayuda: libros en pantalla, ayuda en pantalla y ayuda interactiva. Permite la interacción con internet, con la herramienta Visual Basic Script que toma el control sobre los contenidos de una página WEB. Se puede también insertar un control Active X en una página WEB con Microsoft Active X Control Pad, que permite la introducción de manuscritos en Visual Basic.

#### 3.6.8.4 Comparación de Productos Front-End.

Tomamos en consideración los Lenguajes de programación descritos anteriormente, presentamos un cuadro descriptivo del análisis de los productos:

	Delphi 5.0	Visual Basic 6.0	Power Buidler 7.0.1
Conversión de tipos (tipado)	SI	NO	NO
Lenguaje Estructurado	SI	SI	SI
Componentes Activos X	SI	SI	SI
Depuración y control de errores	ALTO	ALTO	ALTO
Control de gráficos	EXCELENTE	BUENO	BUENO
Opciones multimedia	SI	SI	SI
Manejo de bases de datos, interfaz	BIEN	EXCELENTE	BIEN
Aprendizaje necesario	REGULAR	REGULAR	SENCILLO
Elaboración de pantallas	SENCILLO	SENCILLO	REGULAR
Desempeño de los drivers que utilice para la conexión a la base de datos.	BUENO	BUENO	BUENO
Facilidades para manejo de interfaz usuario	SI	SI	SI
Facilidades para que el programador	Data Access Data Control		Transaction Objects

<b>pueda controlar las conexiones a la base de datos</b>			
<b>Utilización de controles</b>	VBX u OCX	VBX y OCX	VBX y OCX
<b>Soporte por parte del proveedor</b>	EXCELENTE	BUENO	BUENO
<b>Costos</b>	BAJO	BAJO	BAJO
<b>Tiempo en segundos para traer 100 registros de una tabla</b>	1.5 Usando BDE 18.5 Usando ODBC	2.9	2.8
<b>Soporte para Multiplataforma</b>	NO	SI	SI
<b>Ventaja principal</b>	Aplicativos 100% ejecutables Genera DLL estándar de Windows lo cual disminuye el tamaño y requerimientos de memoria	La integración con las base de datos e internet	Los Datawindows

Figura 3.6.8.4.1 Cuadro comparativo entre productos para realizar el front-end<sup>6</sup>

### 3.6.9 Elección de la óptima solución

En la búsqueda de soluciones para resolver la problemática planteada por la Empresa hipotecaria se tomó en cuenta, además del desempeño, la viabilidad de la opción que se pretende implantar. En consecuencia, se contemplaron las expectativas que se tienen del producto final, así como la situación económica de la empresa, la relación costo beneficio, la experiencia de los desarrolladores, tiempo de implantación, entre otras consideraciones.

#### La compra de un producto

Para evaluar dicha opción se tuvo que considerar los pasos subsecuentes a seguir, como son el porcentaje de los objetivos que serían cubiertos por dicho sistema, la

<sup>6</sup> Ricardo Gómez (Investigador CIFI-Informática Universidad de los Andes), DEA Informática, Francia 1992

factibilidad de subsecuentes modificaciones, los costos de compra, asesoría y consultoría así como de su implantación.

### **Adecuación del sistema actual**

En realidad no existe un sistema como tal, pues la información como se mencionó en las secciones anteriores se maneja a través de hojas de cálculo, lo cual resulta inadaptable ante las exigencias de la empresa.

Creación de un sistema a la medida.

Esta opción resulta más atractiva, pues puede cubrir de manera más efectiva los requisitos de la empresa y respondería adecuadamente a los diferentes escenarios planteados por la misma, con un costo mucho menor que al recurrir a una consultoría externa.

De acuerdo al análisis anterior de las opciones disponibles tanto para la construcción del backend como del frontend, que se utilizarán para crear tal sistema, se determinó lo siguiente:

#### **3.6.9.1 Elección del Back-End**

Para la selección del Back-End, se analizaron algunos factores que facilitaron la elección:

- Facilidad para el uso de la herramienta
- Rapidez en la aplicación
- Integración del RDBMS con la plataforma Windows
- Componentes y costo

Tomando en cuenta el cuadro comparativo de la figura 3.6.7.4.2, se optó finalmente por SQL Server de Microsoft, ya que cumple con los requerimientos para el manejo de una base de datos relacional y que además presenta la gran ventaja de que puede ser escalable en cuanto al manejo de una pequeña base con pocos usuarios, hasta una base del orden de TB (Terabyte) con una gran cantidad de usuarios, lo que la hace ser competitiva con manejadores como Oracle Y MySQL que tradicionalmente han sido las herramientas mas robustas para manejo de grandes cantidades de información, pero con precios muy por encima de los que ofrece SQL Server, en los aspectos de planes de capacitación, contratos de mantenimiento así como en los esquemas de licenciamiento.

### **3.6.9.2 Elección del Front-end**

Para una adecuada elección del front-end se deben seleccionar las herramientas de desarrollo adecuadas para poder cumplir con las expectativas del usuario.

Se debe asegurar que las necesidades del usuario sean cubiertas de manera general, es decir, la solución no solo debe satisfacer las necesidades de un determinado grupo de usuarios. También deben de tomarse en cuenta los siguientes puntos antes de elegir la solución de la óptima.

#### **Tecnología**

El pensar en desarrollar un sistema de tal magnitud en una tecnología que ya es obsoleta o que en poco tiempo lo será es una decisión errada. Al pensar en tecnologías nuevas sabemos que se encontraran las herramientas en el mercado con nuevas innovaciones, además de que se contara en todo momento con personal capaz de dar mantenimiento a nuestro sistema.

#### **Herramientas de apoyo**

El iniciar el desarrollo de un nuevo sistema, no quiere decir que se tenga que empezar de cero, siempre en el mercado existen lenguajes de programación y manejadores de bases de datos que nos ofrecen un sin número de ayudas que facilitaran nuestro trabajo y sobre todo acortaran los tiempos de desarrollo. Es importante recalcar que nuestra herramienta con la cual vamos a construir el front-end debe de contar con funciones ya construidas, además de manejo de ventanas y menús a fin de evitar que seamos nosotros los que desarrollemos estas utilerías.

Por lo anterior, además porque la empresa cuenta ya con una licencia, nuestra opción fue Power Builder, por la facilidad que ofrece en su utilización, la opción de utilizar Data Windows, su facilidad para interactuar con las bases de datos y además porque es multiplataforma, adaptable a cualquier sistema y cuenta con muchos componentes que pueden facilitar la aplicación.





## CAPÍTULO IV DESARROLLO E IMPLANTACIÓN DEL SISTEMA

### 4.1 Aplicación de la metodología elegida (Backend)

En el capítulo 2, dentro del tema 2.3, pudimos conocer la metodología para las bases de datos relacionales, en donde se explicó, de manera detallada, la metodología de Yourdon, misma que utilizaremos en este caso.

Dentro de este capítulo, se realizará un análisis de forma estructurada, el cual corresponde al desarrollo del sistema para "Automatizar el Proceso de Cobranza de una Empresa Hipotecaria". Continuando con la metodología de Yourdon, las técnicas que se utilizarán para tal efecto son las que se mencionan a continuación:

- Diagrama de Contexto.
- Diagrama de Flujo de Datos.
- Diccionario de Datos.
- Diagrama Entidad-Relación.
- Normalización.

#### 4.1.1 Diagrama de Contexto

El Diagrama de Contexto es aquel que nos proporcionará, de manera gráfica, la forma en la que los datos son manejados y controlados.

Dentro del primer nivel de un Diagrama de Flujo de Datos encontramos el Diagrama de Contexto, en donde, a través de un diagrama sencillo, se engloba y se representa el sistema por completo, considerando los datos de entrada y de salida. Cabe señalar que el Diagrama de Contexto resalta las características importantes del sistema:

## CAPÍTULO IV DESARROLLO E IMPLANTACIÓN DEL SISTEMA

### 4.1 Aplicación de la metodología elegida (Backend)

En el capítulo 2, dentro del tema 2.3, pudimos conocer la metodología para las bases de datos relacionales, en donde se explicó, de manera detallada, la metodología de Yourdon, misma que utilizaremos en este caso.

Dentro de este capítulo, se realizará un análisis de forma estructurada, el cual corresponde al desarrollo del sistema para "Automatizar el Proceso de Cobranza de una Empresa Hipotecaria". Continuando con la metodología de Yourdon, las técnicas que se utilizarán para tal efecto son las que se mencionan a continuación:

- Diagrama de Contexto.
- Diagrama de Flujo de Datos.
- Diccionario de Datos.
- Diagrama Entidad-Relación.
- Normalización.

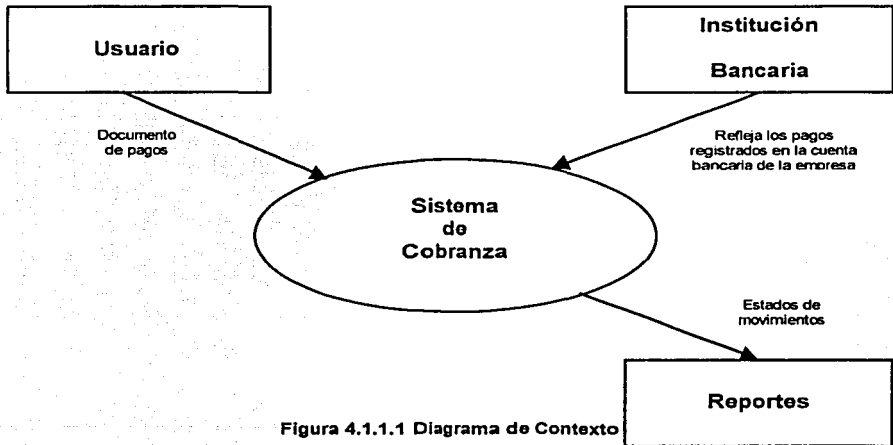
#### 4.1.1 Diagrama de Contexto

El Diagrama de Contexto es aquel que nos proporcionará, de manera gráfica, la forma en la que los datos son manejados y controlados.

Dentro del primer nivel de un Diagrama de Flujo de Datos encontramos el Diagrama de Contexto, en donde, a través de un diagrama sencillo, se engloba y se representa el sistema por completo, considerando los datos de entrada y de salida. Cabe señalar que el Diagrama de Contexto resalta las características importantes del sistema:

- Entidades externas, las cuales son los usuarios o los subsistemas con los que interactúa el sistema.
- Los Datos que son recibidos por el sistema, por medio de las entidades externas y los cuales son recibidos de su entorno y, además, la manera en que deben procesarse en el mismo.
- Los Datos que son producidos por el sistema y los cuales son enviados al exterior.
- Los almacenes que produce el sistema y donde son guardados los datos y que posteriormente se envían al exterior.
- La frontera existente entre el sistema y el exterior.

En la figura 4.1.1.1 se muestra el diagrama de contexto, del cual podemos decir que es el nivel 0 del diagrama de flujo de datos. En él podemos observar que el sistema se comunica con cuatro entidades, las cuales son las encargadas de proveer de información al mismo. Esta información es procesada y, posteriormente, se generan los reportes de salida, los cuales son muy útiles para algunas de las entidades y principalmente para el usuario.



## **Usuario**

Existe un encargado de cobranza, que es quien recibe los pagos por diversos conceptos y valida las firmas, por último esta persona o personas capturan los datos pertinentes para introducirlos en el sistema.

## **Bancos**

La Línea Empresarial es el vínculo existente entre la Institución Bancaria y la Empresa Hipotecaria, misma que proporciona el estado que presenta la cuenta de la empresa.

## **Reportes**

El sistema produce datos de carácter informativo para los usuarios, a los cuales denominamos reportes.

### **4.1.2 Diagrama de Flujo de Datos**

El Diagrama de Flujo de Datos es una técnica gráfica que representa el flujo de la información y las transformaciones que se aplican a los datos al moverse desde la entrada hasta la salida. Este diagrama se puede usar para representar un sistema o un software a cualquier nivel de abstracción. Los diagramas de flujo de datos pueden ser refinados en niveles que representen un mayor flujo de información y un mayor detalle funcional.

#### **Diagrama de flujo de datos a nivel 1**

A partir del diagrama de contexto o diagrama de flujo de datos a nivel 0, se genera el diagrama de datos a nivel 1, en el cual cada uno de los procesos representados, son una subfunción del sistema general en el nivel anterior.

En la figura 4.1.2.1 se muestra como las entidades externas envían información al sistema y después, éstos son procesados por él mismo y a su vez genera uno o varios reportes que ayudarán al usuario.

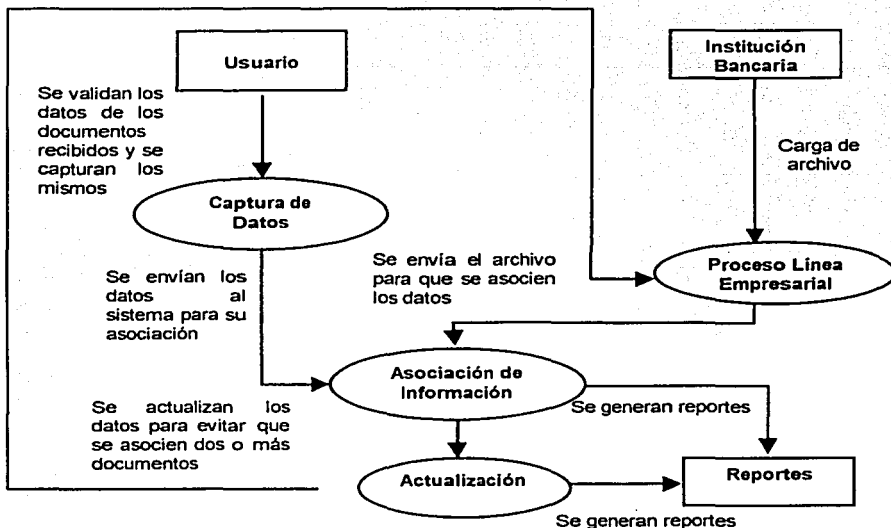


Figura 4.1.2.1 Diagrama de Flujo de Datos

### Módulo Asociación de Información

En este módulo se relaciona la información obtenida de la captura de los datos de los documentos recibidos por el usuario, con el archivo recibido de la Línea Empresarial y de esta manera se checan los documentos que se tienen como cobrados, generando un reporte de la información. Además este módulo asigna un estatus a los pagos recibidos para saber si han sido cobrados, están pendientes o han sido rechazados.

### Módulo de Proceso Línea Empresarial

Este módulo se encarga de procesar la información directamente recibida de la Línea Empresarial y de la actualización proporcionada por el sistema para que no exista la posibilidad de asociar dos o más pagos con el Intermediario Financiero, y así llevar un mejor control de la cobranza.

### Módulo de Actualización

Es el encargado de, valga la redundancia, actualizar los datos y las asociaciones previamente realizadas, lo cual evitará que se repitan las asociaciones de los documentos con el archivo proveniente de la institución bancaria, a través de la Línea Empresarial.

### Diagrama De Flujo de Datos a Nivel 2

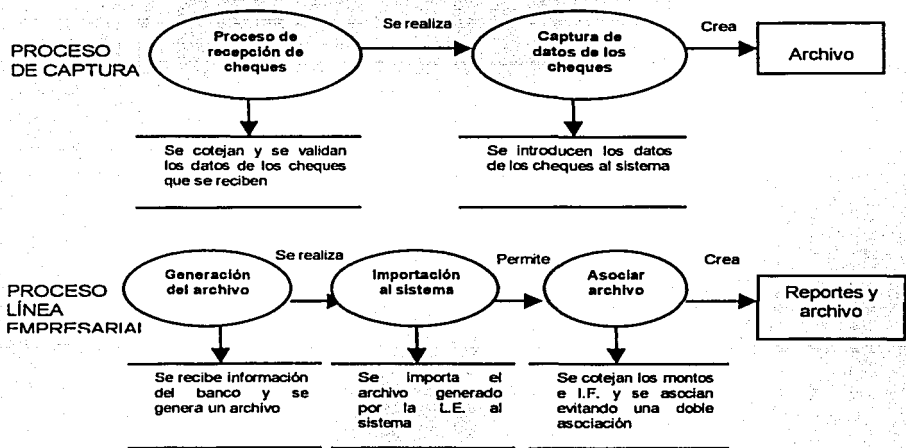


Figura 4.1.2.2 Diagrama de Flujo de Datos a Nivel 2

### **Módulo de Captura de Datos**

En este punto es donde el usuario interactúa directamente con el sistema, ya que aquí es donde él introduce los datos propios y pertinentes de los documentos previamente recibidos y validados (en cuanto a la firma autorizada y el Intermediario Financiero). Una vez capturados los datos, éstos son asociados con su respectivo Intermediario Financiero y los montos pagados, tal como se muestra en la figura 4.1.2.2.1. En esta interacción del usuario con la información que proporciona al sistema, los datos se validan sabiendo si existe el Intermediario Financiero, entre otros datos, y posteriormente éstos serán importados al sistema para ser asociados más adelante.

### **Módulo de Generación del Archivo**

En este módulo se genera información a partir de los datos recibidos en una computadora conectada directamente a la Institución Financiera, con los datos de los pagos que han sido efectuados por parte del Intermediario Financiero, ya sea vía cheques, depósitos u otro medio, los cuales serán asociados por el sistema con el archivo previamente generado por el Módulo de Captura de Datos. Esto nos permite saber los Intermediarios que han realizado sus pagos, sus montos y posteriormente nos enteraremos si queda algún remanente por cobrar.

### **Módulo de Login y Password**

En este paso el usuario introduce su login y su password. Éste es parte del proceso de captura de datos que vimos en la Figura 4.1.2.2.1, para ver las opciones a las que tiene acceso el usuario y si puede o no modificarlos. Dependiendo de los accesos a que tiene derecho, al usuario se le permite la introducción de datos o modificación de los mismos o simplemente una consulta de los pagos que han sido recibidos.



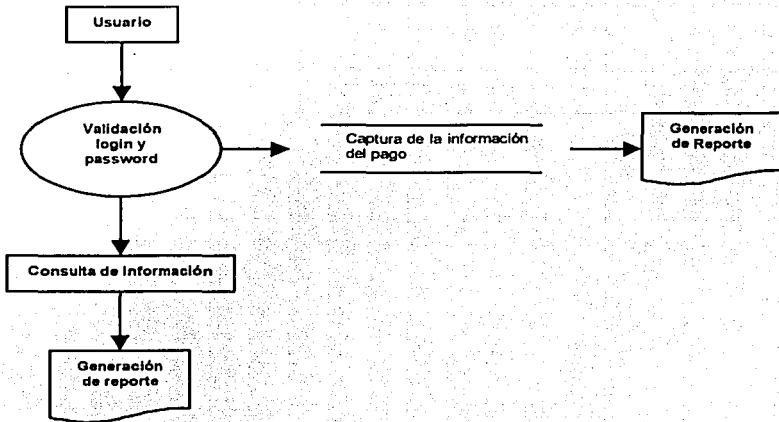


Figura 4.1.2.3 Diagrama de Flujo de Datos a Nivel 3

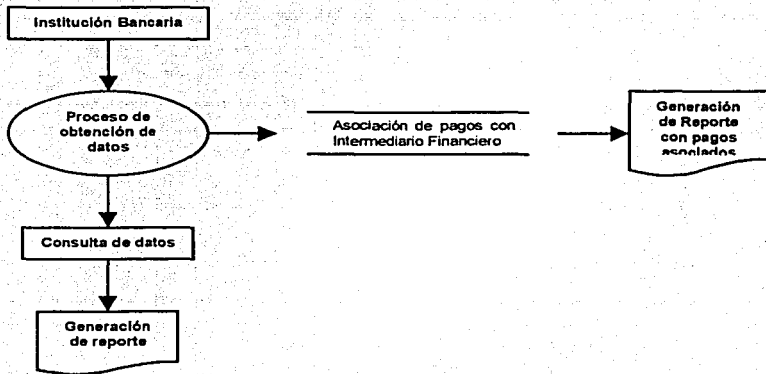


Figura 4.1.2.4 Diagrama de Flujo de Datos a Nivel 3

### Módulo de obtención de datos

En este proceso se reciben los datos de la Línea Empresarial y se generan un reporte y un archivo. El reporte nos sirve para ver los pagos realizados y el archivo es muy útil para el sistema, ya que con este archivo se procede a la asociación de los pagos con los documentos recibidos del Intermediario Financiero.

### Módulo Importación de Datos

Este módulo es el encargado de registrar los datos generados a partir de la información de la Línea Empresarial al sistema para que sean procesados por el mismo y se lleve a cabo la asociación de pagos.

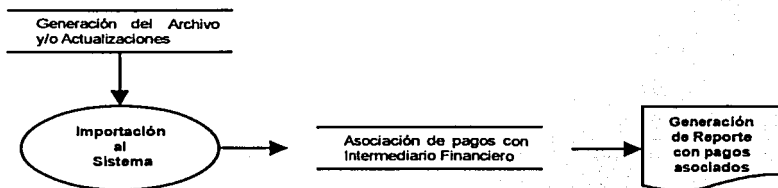


Figura 4.1.2.5 Diagrama de Flujo de Datos a Nivel 3

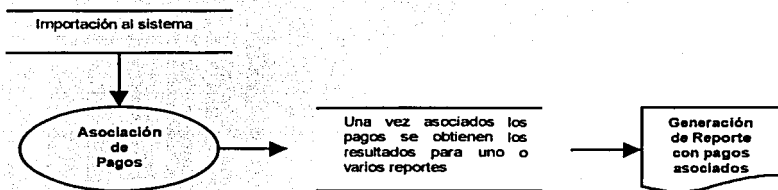


Figura 4.1.2.6 Diagrama de Flujo de Datos a Nivel 3

### 4.1.3 Diccionario de Datos

El propósito de un diccionario de datos es documentar de una manera adecuada todas y cada una de las instancias que conforman la base de datos a utilizar, permitiendo así tener una referencia completa de los atributos de la Base de datos. Ya que el diccionario de datos es una referencia útil para localizar nombres y atributos de los datos en el sistema, éste deberá actualizarse de manera oportuna cuando se realicen cambios al sistema. La figura 4.1.3.1 muestra las tablas que integran nuestro Sistema de Cobranza.

Nombre	Acrónimo
Catálogo de Conceptos	CAT_CONCEPTOS
Catálogo de Contactos de Intermediarios Financieros	CAT_CONTACTO_IF
Catálogo de Estatus	CAT_STATUS
Catálogo de Motivos de Rechazo	CAT_MOTIVO_RECHAZO
CETES	CETES
Departamento	CAT_DEPTO
Documentos por Cobrar	DOCXCOBRAR
PAGOS	PAGOS
Tabla de Usuarios de Sistema	USUARIO
Menús de Sistema	MENUS
Permisos	USUARIOSACCESO
Archivo de Línea Empresarial	LINEALEMPRESARIAL
Pagos Asociados	ASOCIAPAGOS
Catálogo de Tipo de Pagos	CAT_TIPO_PAG

TABLA 4.1.3.1 Tablas de Sistema

En las figuras subsecuentes mostraremos el diccionario de datos de cada una de las tablas que integran el sistema. En dichas figuras se podrá observar un Acrónimo, que es el nombre clave del campo, debe de expresarse en códigos sencillos, fáciles de relacionar con el dato contenido. También se incluye la descripción del campo, el tipo de dato que va a contener, la longitud del dato y a que tipo de atributo nos estamos refiriendo.

Catálogo de Conceptos (CAT_CONCEPTOS)				
Acrónimo	Descripción	Tipo	Longitud	Tipo de Llave
CVE_CONCEPTO	Clave de Concepto	Smallint	2	PK
DESCRIPCION	Descripción	Char	30	

TABLA 4.1.3.2

Catálogo de Contactos de Intermediarios Financieros (CAT_CONTACTO_IF)				
Acrónimo	Descripción	Tipo	Longitud	Tipo de Llave
CVE_IF	Clave de Intermediario Financiero	Char	3	PK
NOMBRE_IF	Nombre del Intermediario Financiero	Char	50	
CONTACTO	Contacto	Char	50	
TELEFONO	Teléfono	Char	14	

TABLA 4.1.3.3

Catalogo de Estatus (CAT_STATUS)				
Acrónimo	Descripción	Tipo	Longitud	Tipo de Llave
CVE_STATUS	Estatus	Char	2	PK
ST_DESCRIPCION	Descripción	Char	30	

TABLA 4.1.3.4

Catálogo de Motivos de Rechazo (CAT_MOTIVO_RECHAZO)				
Acrónimo	Descripción	Tipo	Longitud	Tipo de Llave
CVE_MOTIVO	Estatus	Smallint	2	PK
DESCRIPCION	Descripción	Char	30	

TABLA 4.1.3.5

La tabla de Cetes, mostrada en la figura 4.1.3.6 va a contener el valor del CETE que se utilizará para el cargo de intereses moratorios.

CETES (CETES)				
Acrónimo	Descripción	Tipo	Longitud	Tipo de llave
FECHA	Fecha de Aplicación del Valor de CETES	Date	8	
VALOR	Valor del CETE	Money	30	

TABLA 4.1.3.6

Departamento (CAT_DEPTO)				
Acrónimo	Descripción	Tipo	Longitud	Tipo de llave
CVE-DEPTO	Clave del Departamento	Char	2	PK
NOMBRE_DEPTO	Nombre de Departamento	Char	55	

TABLA 4.1.3.7

Documentos por Cobrar (DOCXCOBRAR)				
Acrónimo	Descripción	Tipo	Longitud	Tipo de llave
NO_MOVI	Consecutivo de Movimiento	Int	4	PK
CVE_MOTIVO	Clave de Motivo	Smallint	2	FK
DIAS_TRANS	Días Transcurridos	Smallint	2	
MONTO	Total Adeudado	Money	8	

TABLA 4.1.3.8

PAGOS (PAGOS)				
Acrónimo	Descripción	Tipo	Longitud	Tipo de llave
CVE_TIP_PAG	Clave tipo Pago	Smallint	2	FK
CVE_CONCEPTO	Clave de Concepto	Smallint	2	FK
CVE_IF	Clave de Intermediario Financiero	Char	3	FK
CVE_STATUS	Estatus	Char	2	FK
USU_CVE_USUARIO	Clave de Usuario	Varchar	8	FK
FOLIO	FOLIO	Char	16	PK
MONTO	Total de Pago	Money	15,2	
FECHA_RECEPCION	Fecha de Recepción	Date		
FECHA_DOCUMENTO	Fecha de Documento	Date		
NO_MOVI	Referencia de DOCXCOB	Int	4	FK <NULL>
FOLIO_PAGO	Folio de registro de Pago asociado	Int	4	FK <NULL>

TABLA 4.1.3.9

Usuarios del Sistema (USUARIO)				
Acrónimo	Descripción	Tipo	Longitud	Tipo de llave
CVE_USUARIO	Clave de Usuario	Varchar	8	PK
PASSWORD	Clave de acceso	Varchar	10	
CVE_DEPTO	Departamento	Char	3	
NOMBRE	Nombre del Usuario	Varchar	50	
CVE_STATUS	Estado	Char	2	

TABLA 4.1.3.10

Tipo de Pago (CAT_TIPO_PAG)				
Acrónimo	Descripción	Tipo	Longitud	Tipo de llave
CVE_TIP_PAG	Clave de tipo Pago	Smallint	2	PK
DESCRIPCION	Descripción	Char	30	

TABLA 4.1.3.11

Linea Empresarial (LINEA_EMPRESARIAL)				
Acrónimo	Descripción	Tipo	Longitud	Tipo de llave
FOLIO_LINEA	Consecutivo de Carga	Int	4	PK
FEC_EDOCTA	Fecha del Estado de Cuenta (LIEM)	Date		
CVE_MOV	Clave del movimiento	Varchar	5	
DESCRIPCION_MOV	Descripción del Movimiento	Varchar	50	
MONTO_CARGO	Monto de Cargo aplicado	Money	8	
MONTO_ABONO	Monto del Crédito aplicado	Money	8	
SALDO_CTA	Total en Saldo de la cuenta	Money	8	
CVE_MOVEDO	Clave del estado del Movimiento	Varchar	5	
FOLI_TRANSFER	Folio asignado a SPEUAS	Varchar	5	
OPERADOR	Operador que realizo la transacción	Varchar	20	
REFERENCIA	Referencia	Varchar	50	
CVE_SUCURSAL	Sucursal donde se realizo el movimiento	Varchar	5	
BCO_ORIGEN	Banco de Origen	Varchar	3	
CVE_DEV	Clave de Devolución	Varchar	2	
HORA	Hora de la transacción	Varchar	6	
CVE_STATUS	Estatus del Pago.	Char	2	

TABLA 4.1.3.15





### 4.1.5 Normalización.

La normalización es la simplificación de las relaciones existente entre los datos, las aplicaciones y los almacenamientos utilizados por los usuarios; de tal manera que se establezcan estructuras de datos de menor tamaño. Para realizar el proceso de Normalización partiremos de dos fuentes importantes de datos, dentro de nuestro proceso de Cobranza: El archivo en Excel que elabora el departamento de Cobranza con la recopilación de información diaria recibida se muestra en la figura 4.1.5.1.

Entidad	Archivo de Cobranza.
Número de Cheque	Folio de documento que sé esta recibiendo
Nombre del Intermediario Financiero	Nombre del intermediario Financiero que esta realizando el pago
Monto	Monto del Documento recibido
Tipo de Pago	Como se esta realizando el pago, a través de un CHEQUE, de un Deposito Bancario o un SPEUA
Concepto Pagado	Cual es el motivo del pago: MESUALIDAD, INTERES MORATORIOS, PENALIZACION, CUOTA de APERTURA entre otros.
Fecha de Recepción	Fecha en la cual se recibe el documento
Fecha del documento	Fecha en la cual se hace efectivo el documento
Contacto	Nombre del contacto del intermediario financiero
Teléfono	Teléfono de Contacto
Analista	Nombre del usuario que dará el seguimiento
Departamento	Departamento del usuario que dará el seguimiento

Figura. 4.1.5.1 Relación de Pagos recibidos

La relación de Pagos posee múltiples valores para el atributo Intermediario Financiero, el cual es un identificador único y por lo tanto no es una forma plana. Para cumplir con la primera forma norma es necesario tener los datos en forma plana. Considerando lo anterior, la conversión de la relación a una forma plana se muestra en la figura 4.1.5.2. La redundancia de los registros se redujo separando los atributos en otra entidad, debiendo asignar un identificador de la entidad original para mantener la relación. En

### 4.1.5 Normalización.

La normalización es la simplificación de las relaciones existente entre los datos, las aplicaciones y los almacenamientos utilizados por los usuarios; de tal manera que se establezcan estructuras de datos de menor tamaño. Para realizar el proceso de Normalización partiremos de dos fuentes importantes de datos, dentro de nuestro proceso de Cobranza: El archivo en Excel que elabora el departamento de Cobranza con la recopilación de información diaria recibida se muestra en la figura 4.1.5.1.

Entidad	Archivo de Cobranza.
Número de Cheque	Folio de documento que sé esta recibiendo
Nombre del Intermediario Financiero	Nombre del intermediario Financiero que esta realizando el pago
Monto	Monto del Documento recibido
Tipo de Pago	Como se esta realizando el pago, a través de un CHEQUE, de un Deposito Bancario o un SPEUA
Concepto Pagado	Cual es el motivo del pago: MESUALIDAD, INTERES MORATORIOS, PENALIZACION, CUOTA de APERTURA entre otros.
Fecha de Recepción	Fecha en la cual se recibe el documento
Fecha del documento	Fecha en la cual se hace efectivo el documento
Contacto	Nombre del contacto del intermediario financiero
Teléfono	Teléfono de Contacto
Analista	Nombre del usuario que dará el seguimiento
Departamento	Departamento del usuario que dará el seguimiento

Figura. 4.1.5.1 Relación de Pagos recibidos

La relación de Pagos posee múltiples valores para el atributo Intermediario Financiero, el cual es un identificador único y por lo tanto no es una forma plana. Para cumplir con la primera forma norma es necesario tener los datos en forma plana. Considerando lo anterior, la conversión de la relación a una forma plana se muestra en la figura 4.1.5.2. La redundancia de los registros se redujo separando los atributos en otra entidad, debiendo asignar un identificador de la entidad original para mantener la relación. En

este caso se separó la información de Intermediarios Financieros, generando así una nueva entidad y asignaremos su llave Clave de intermediario para mantener la liga con los pagos recibidos.

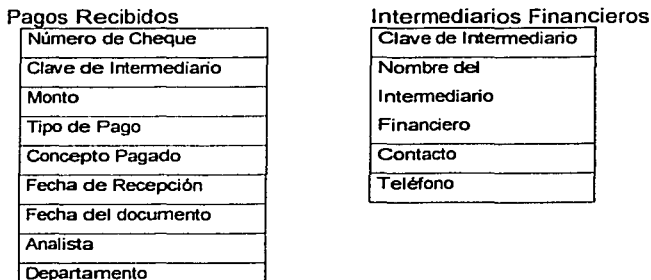


Figura. 4.1.5.2 Relación de Pagos recibidos e Intermediarios Financieros

La prueba para determinar si una entidad está en 2FN es: que el valor de cualquier atributo que no es llave, dependa de todos los atributos que forman la llave. Para normalizar una entidad a 2FN, se crea una nueva entidad de los atributos que dependen parcialmente de una llave, siendo así parte del identificador de esta la nueva entidad el atributo del cual depende para mantenerla relacionada con la original.

En la figura 4.1.5.3, tenemos la información del concepto de pago que no depende directamente del folio, el tipo de pago y el Analista, que en adelante llamaremos usuario. Para normalizar nuestra entidad a la 2FN, se crean 3 nuevas entidades: **Conceptos de pagos, Tipos de Pagos y Usuarios** (analistas).

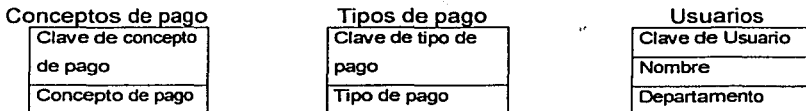


Figura. 4.1.5.3 Relación de conceptos de pago, tipos de pago y usuarios

Al obtener las anteriores entidades, nuestra información de cobranza quedaría como lo muestra la figura 4.1.5.4. La entidad de Pagos ahora cuenta con los atributos Clave de tipo de Pago, Clave de concepto de pago y clave de usuario con lo que se mantiene la liga con las nuevas entidades derivadas de la 2FN.

Número de Cheque(Folio)
Clave de Intermediario
Monto
Clave de tipo de Pago
Clave de Concepto pagado
Fecha de Recepción
Fecha del Documento
Clave de Usuario

**Figura. 4.1.5.4 Relación de Pagos recibidos**

Una entidad está en FN3 si el valor de cada atributo depende de toda la llave y no de cualquier otro que no lo sea. Analizando nuestra entidad de PagosRecibidos que muestra la figura 4.1.5.4, podemos ver que no existe ningún atributo que no depende de la llave FOLIO, por lo cual nuestra entidad está en 3FN.

## 4.2 Construcción del Back-End

### 4.2.1 Diseño del Back-End

Para el diseño de **Back-End**, nos apoyamos en una herramienta de diseño de Base de Datos llamada PowerDesigner AppModeler. El PowerDesigner es una herramienta robusta de diseño, para aplicaciones en Cliente/Servidor. Con éste, se puede construir un Modelo de Datos Físico (PDM) para un DBMS<sup>1</sup>, (un PDM representa gráficamente, la interacción entre los diversos objetos de la Base de Datos, como son Tablas, columnas, llaves, vistas y otros); además nos permite generar el Script de la Base de datos para la generación física de la misma, poniendo cuidado en la integridad referencial<sup>2</sup>. Permite imprimir los modelos en diversos tipos de gráficos, aplicar reingeniería inversa al modelo, así como también definir atributos extendidos para aplicaciones 4GL<sup>3</sup>.

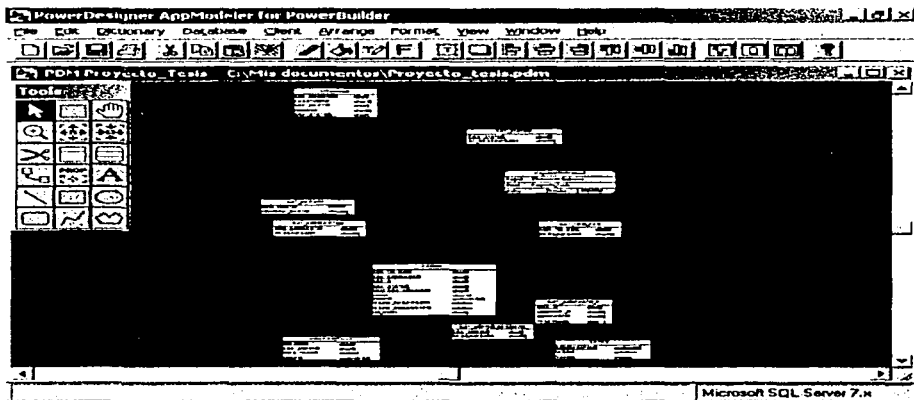


Figura 4.2.1.1. PowerDesigner, DER del BackEnd.

<sup>1</sup> DBMS Sistema gestor de Base de Datos

<sup>2</sup> Integridad Referencial,

<sup>3</sup> 4GL Aplicaciones de Cuarta Generación.

La pantalla de PowerDesigner luce como se muestra la figura 4.2.1.1, en la cual se pueden observar las diversas entidades que conforman la Base de Datos; sus llaves primarias y foráneas, sus atributos y características, y de las relaciones existentes entre ellas. PowerDesigner, al igual de PowerBuilder nos permite conectarnos a diversos DBMS, utilizando las conexiones de ODBC, JDBC y OLE entre otros, permitiendo así un desarrollo fácil y rápido. También permite la generación de Procedimientos almacenados, Triggers<sup>4</sup> y Queries<sup>5</sup>, entre otros. En la figura 4.2.1.2, se muestra la opción de DataBase. Dentro de ésta opción se realiza la conexión y configuración al Servidor de SQL, se definen algunas características de los archivos físicos de datos y logs<sup>6</sup>, y las opciones de Generación de Bases de Datos, Triggers, Procedimientos Almacenados, además de la posibilidad de generar datos de prueba para la base de datos.

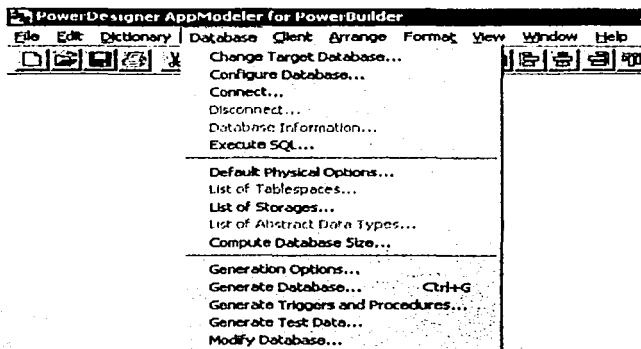


Figura 4.2.1.2. Opciones de Base de Datos en PowerDesigner.

Una Base de Datos se genera a partir de un Modelo Gráfico, como el mostrado en la figura 4.2.1.1. Las tablas se definen con el símbolo de Entidad del DER, y se les asigna un nombre clave y una descripción. La figura 4.2.1.3 muestra la pantalla de generación de la tabla USUARIO. En la parte inferior de ésta, se puede apreciar los

<sup>4</sup> Triggers Disparadores

<sup>5</sup> Queries Consultas a Base de Datos

<sup>6</sup> Logs, registros de actividad

botones de generación de Columnas, Indices, Llaves Alternas, Triggers entre otros, que permiten, de forma sencilla, la generación de los diferentes atributos que contendrá la tabla. También se puede añadir una descripción más detallada, si utilizamos la pestaña de Descripción.

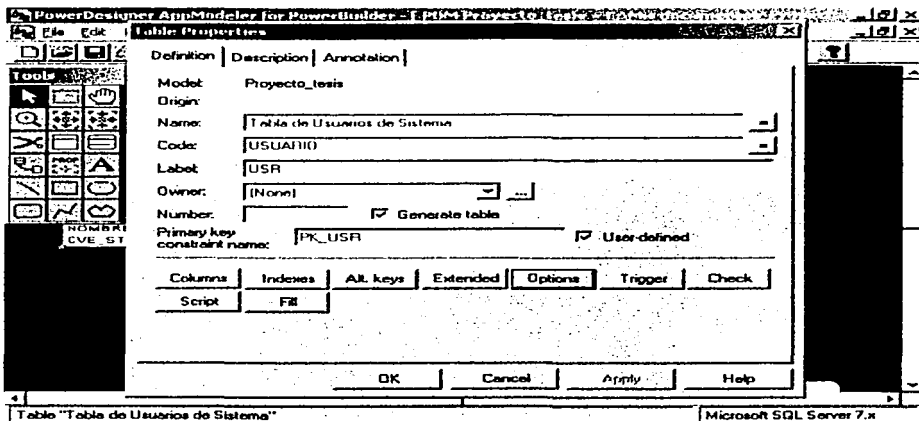


Figura 4.2.1.3 Construcción de la Tabla USUARIO.

Nos enfocaremos únicamente a la generación de los atributos, ya que para esta tabla no se han definido Triggers. Seleccionamos el Botón de Columns<sup>7</sup> para definir los campos o atributos que contendrá nuestra tabla. Como lo muestra la figura 4.2.1.4, desde esta opción podemos definir del atributo: Una descripción general (máxima 50 caracteres), el nombre clave o acrónimo (máximo 78 caracteres), tipo (permite diversos tipos de datos como char<sup>8</sup>, varchar<sup>9</sup>, int<sup>10</sup>, decimal, money<sup>11</sup>, entre otros), longitud (depende de cada tipo de dato), si el atributo es llave primaria, si el valor puede o no ser nulo, y si es desplegable. También es posible establecer un dominio - conjunto de valores válidos - para el tipo de dato que así lo requiera.

<sup>7</sup> Columns Columns o atributos

<sup>8</sup> Char caracter. De longitud fija

<sup>9</sup> Varchar Caracter de Longitud Variable

<sup>10</sup> INT Numérico, Entero

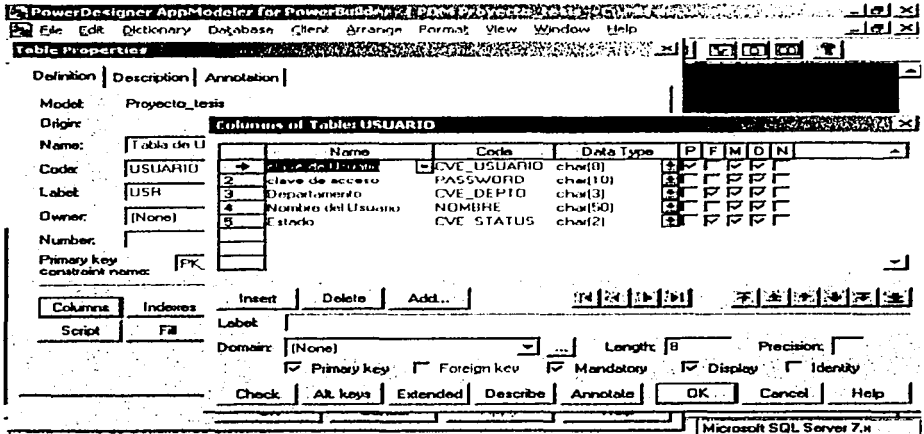


Figura 4.2.1.4 Generación de tabla

De esta forma fácil y sencilla se generaron todas las tablas. Una vez generadas, se procede a la generación de Script. Con la opción "Generate DataBase"<sup>12</sup> del menú DataBase, mostrado en la figura 4.2.1.3, nos presenta la pantalla mostrada en la figura 4.2.1.5 donde se preparan las opciones para la generación de la Base de datos. Aquí podemos elegir los elementos que necesitamos se incluyan en el Script; como incluir un DROP<sup>13</sup> al inicio del Script, para cuando se está redefiniendo la Base de Datos; las sentencias para la creación de Indices, de Vistas y los Tablespace<sup>14</sup>, entre otros. Una vez seleccionadas todas las opciones necesarias se procede a generar la Base de datos o si es necesario a generar el Script.

<sup>11</sup> Money Tipo monetario

<sup>12</sup> Generate DataBase, Genera Base de Datos

<sup>13</sup> DROP Borrado Físico del Objeto.

<sup>14</sup> Tablespace Archivos de Almacenamiento de Base de Datos



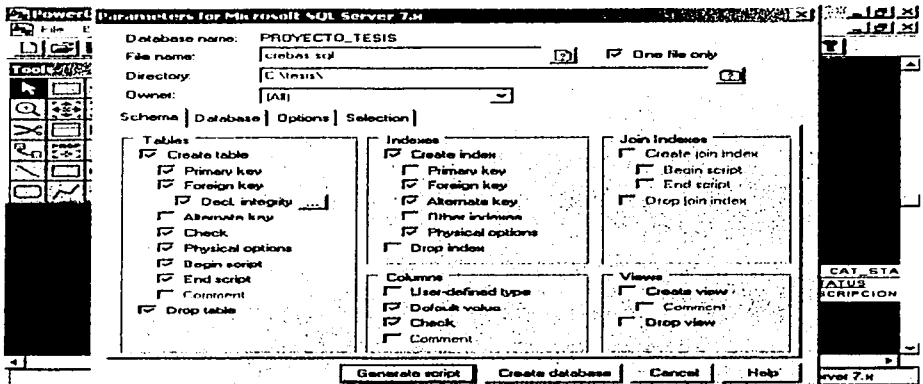


Figura 4.2.1.6 Generación del Script y la Base de Datos.

## 4.2.2 Construcción del BackEnd

Como ya se definió en el capítulo anterior se seleccionó el SQL Server 2000 para la construcción del BackEnd. Para la generación física de la Base de Datos se utilizó el PowerDesigner, el cual generó un Script. La tabla 4.2.2.1 muestra una sección del Script generado.

```

/* ===== */
/* Database name: PROYECTO_TESIS */
/* DBMS name: Microsoft SQL Server 7.x */
/* Created on: 15/06/2002 18:43 */
/* ===== */
/* Database name: PROYECTO_TESIS */
/* ===== */
create database PROYECTO_TESIS
go

/* ===== */

```

```
/* Table: CETES */
/* ===== */
create table CETES
(
    FECHA            datetime            not null,
    VALOR            money               not null,
    constraint PK_CETES primary key (FECHA)
)
go
/* ===== */
/* Table: CAT_CONTACTO_IF */
/* ===== */
create table CAT_CONTACTO_IF
(
    CVE_IF           char(3)             not null,
    NOMBRE_IF        char(50)           not null,
    CONTACTO         char(50)           not null,
    TELEFONO         char(14)           not null,
    constraint PK_CCIF primary key (CVE IF)
)
go
```

Tabla 4.2.2.1 Script de la Base de Datos.

Las ventajas de la generación por parte de PowerDesigner, nos permiten no preocuparnos en donde residirán los archivos físicos de datos e índices, porque esto ya fue previamente definido por PowerDesigner. La figura 4.2.2.2 muestra, mediante la Pantalla del SQL Server Enterprise<sup>15</sup>, nuestra Base de datos generada. Aquí podemos observar los diversos objetos que se definieron automáticamente al generarla desde PowerBuilder. Se pueden observar: vistas, tablas, diagramas E-R, procedimientos almacenados; Usuarios y tablas maestra, entre otros.

<sup>15</sup> SQL Server Enterprise, Servidor de SQL Corporativo.

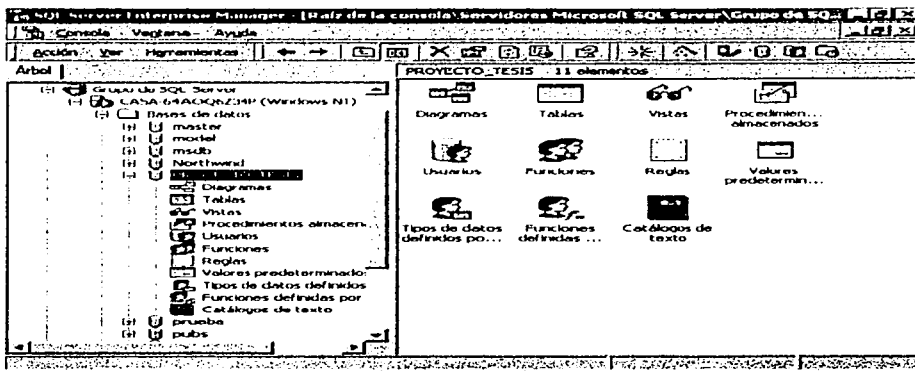


Figura 4.2.2.2 Pantalla del SQL Server Enterprise.

La figura 4.2.2.3 muestra las tablas de la Base de datos. Es posible visualizar el nombre de la tabla, el propietario y la fecha de creación. Desde esta pantalla podemos utilizar, - con click derecho - algunas opciones para tablas que permiten importar y exportar datos, realizar consulta, definir permisos entre otros.

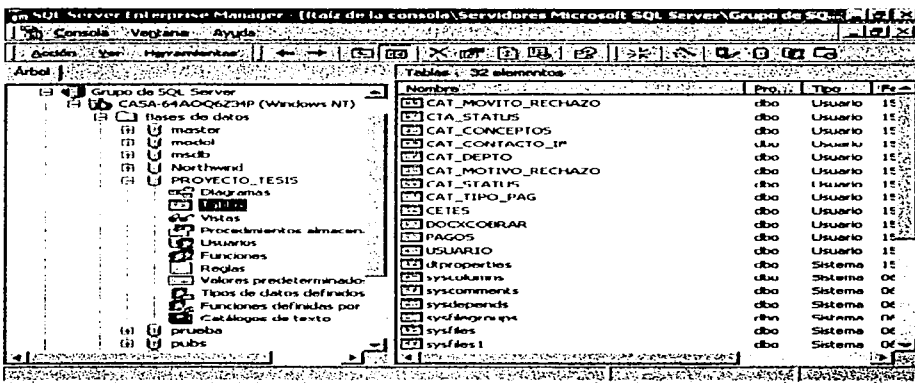


Figura 4.2.2.3 Base de Datos Proyecto de Tesis.

La figura 4.2.2.4 muestra una consulta al Catálogo de Departamentos, esta consulta se realizó a través de la opción de tablas.

CVE_DEPTO	NOMBRE_DEPTO
300	DIRECCION GENERAL
301	SECRETARÍO DE CONSEJO
302	DIRECCION DE RELACIONES INSTITUCIONALES Y DE COMUN
303	SUBDIRECCION DE PROMOCION
304	DELEGACION DISTRITO FEDERAL
305	DELEGACION GUADALAJARA
306	DELEGACION CANCUN
307	DELEGACION TIJUANA
308	DELEGACION MONTERREY
309	DELEGACION VERACRUZ
310	SUBDIRECCION DE RELACIONES INSTITUCIONALES
311	SUBDIRECCION DE COMUNICACIÓN SOCIAL
312	SECRETARIA DE LA DIRECCION
313	CONTRALORIA INTERNA
314	AUDITORIA INTERNA
315	DIRECCION GENERAL ADJUNTA DE SISTEMAS
316	DIRECCION DE TECNOLOGIA DE INFORMACION
317	SUBDIRECCION DE TELECOMUNICACIONES
318	SUBDIRECCION DE SOPORTE A USUARIOS
319	SUBDIRECCION DE COMPLETO
320	DIRECCION DE SISTEMAS DE INFORMACION
321	SUBDIRECCION DE METODOS Y PROCEDIMIENTOS
322	SUBDIRECCION PLANEACION Y DESARROLLO DE SISTEMAS
323	SUBDIRECCION DE SISTEMAS DE FINANZAS Y RIESGOS
324	SUBDIRECCION DE SISTEMAS JURIDICO Y ADMINISTRATIVO
325	SUBDIRECCION DE SISTEMAS DE CREDITO

Figura 4.2.2.4 Tabla Catálogo de Departamento.

### 4.2.3 Seguridad dentro de SQL Server

Existen dos tipos de permisos: sobre Objetos, que restringen el uso de sentencias (DML) tales como INSERT, DELETE, SELECT, UPDATE y EXEC sobre tablas, vistas y procedimientos almacenados; y sobre Comandos, que restringen el uso de sentencias (DDL) de manipulación de objetos tales como CREATE, DROP, ALTER sobre vistas, tablas, procedimientos, Bases de Datos. La figura 4.2.3.1 muestra la generación de un nuevo Usuario. Dentro de esta misma figura se observa la segunda pantalla desplegada para seleccionar el tipo de usuario que se está definiendo.

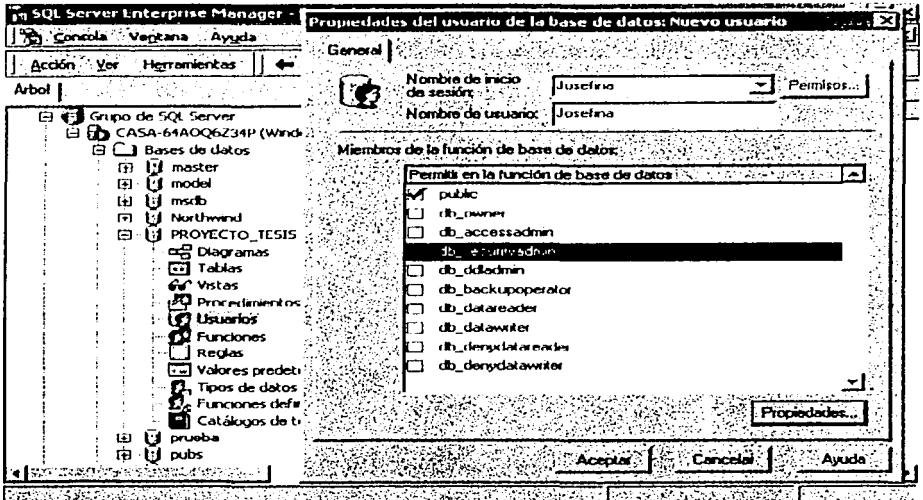


Figura 4.2.3.1 Asignando Permisos desde el SQL Server Enterprise.

El SQL Server Enterprise, nos permite realizar todas las funciones de Administrador de Base de datos, a través de un ambiente gráfico se pueden realizar y programar diversas tareas. La figura 4.2.3.2 nos muestra algunas de las funciones existentes sobre tablas como son:

- Importar y Exportar Datos ya sea desde un archivo de texto, desde un ODBC o desde cualquier origen establecido.
- Realizar copias de seguridad, de forma programada o en el momento.
- Realizar recuperaciones de Bases de Datos, desde diversas copias de seguridad. Permite recuperar la Base de datos sobre sí misma o con un nuevo nombre.
- Permite desconectar la Base de Datos para no permitir su uso.
- Además de otras sub-opciones no visibles.

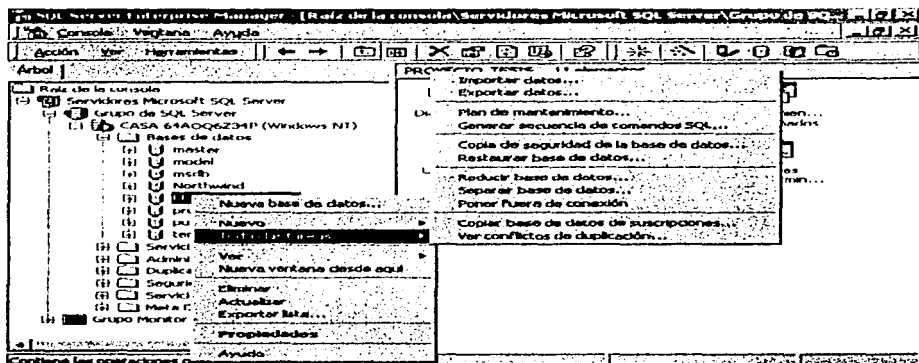


Figura 4.2.3.2 SQL Server Enterprise.

#### 4.2.4 Tamaño de la Base de Datos.

Primero es importante mencionar que SQL Server 2000 utiliza 3 tipos de archivos para almacenar la información:

- **Archivos principales:** Estos archivos contienen la información de inicio para la base de datos. Este archivo se utiliza también para almacenar datos. Cada base de datos tiene un único archivo principal.
- **Archivos secundarios:** Estos archivos contienen todos los datos que no caben en el archivo de datos principal. No es necesario que las bases de datos tengan archivos de datos secundarios si el archivo principal es lo suficientemente grande como para contener todos los datos. Algunas bases de datos pueden ser muy grandes y necesitar varios archivos de datos secundarios o utilizar archivos secundarios en unidades de disco distintas, de modo que los datos estén distribuidos en varios discos.
- **Registro de transacciones:** Estos archivos contienen la información de registro que se utiliza para recuperar la base de datos. Debe haber al menos un archivo de registro de transacciones para cada base de datos, aunque

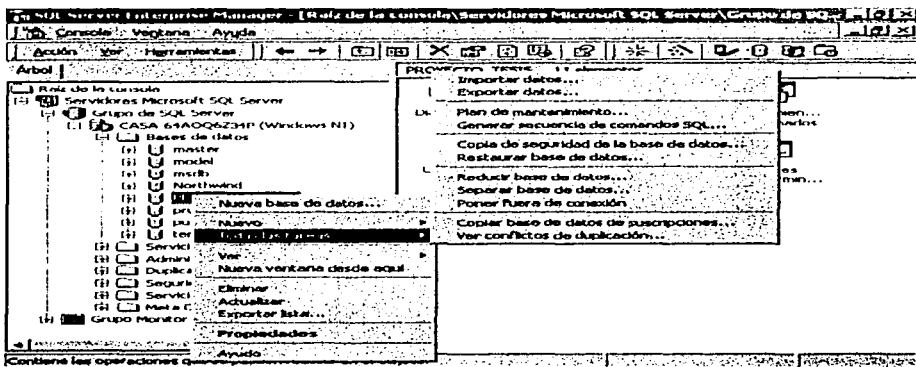


Figura 4.2.3.2 SQL Server Enterprise.

#### 4.2.4 Tamaño de la Base de Datos.

Primero es importante mencionar que SQL Server 2000 utiliza 3 tipos de archivos para almacenar la información:

- **Archivos principales:** Estos archivos contienen la información de inicio para la base de datos. Este archivo se utiliza también para almacenar datos. Cada base de datos tiene un único archivo principal.
- **Archivos secundarios:** Estos archivos contienen todos los datos que no caben en el archivo de datos principal. No es necesario que las bases de datos tengan archivos de datos secundarios si el archivo principal es lo suficientemente grande como para contener todos los datos. Algunas bases de datos pueden ser muy grandes y necesitar varios archivos de datos secundarios o utilizar archivos secundarios en unidades de disco distintas, de modo que los datos estén distribuidos en varios discos.
- **Registro de transacciones:** Estos archivos contienen la información de registro que se utiliza para recuperar la base de datos. Debe haber al menos un archivo de registro de transacciones para cada base de datos, aunque

puede haber más de uno. El tamaño mínimo para un archivo de registro es 512 kilobytes (KB).

Para tener un buen rendimiento en el manejo de una Base de Datos es recomendable realizar un buen diseño de la estructura de la Base de Datos y asignar el espacio primario y el porcentaje de crecimiento adecuados.

Para el cálculo del espacio total de una Base de Datos es necesario conocer 3 aspectos:

- Conocer cual es el espacio utilizado el SQL Server para el manejo de datos.
- Saber el espacio requerido para almacenar cada tipo de datos que se utilizará y,
- El tamaño del bloque del almacenamiento que utiliza el SQL Server.

Las figuras 4.2.4.1 y 4.2.4.2 nos muestran estos aspectos:

Tamaño de Página	8 KB
Tamaño del directorio de datos.	96 bytes

Figura. 4.2.4.1 Datos de almacenamiento de SQL Server.

Tipo de dato	Tamaño en Bytes
Money	8
Smallint	2
Datetime	4
Date	2
Int	4
Char	1 x longitud

Figura. 4.2.4.2 Longitud utilizada por cada tipo de dato.

Ahora analicemos los datos que vamos a almacenar; nuestra Base de datos está compuesta de 15 tablas, de las cuales 7 son catálogos, mismos que después de la carga inicial, tendrán un crecimiento mínimo por la naturaleza de la información que



contienen. Otras tablas tendrán un crecimiento medio, porque la información contenida tiene movimiento más o menos constante. El resto de las tablas, tales como Linea\_empresa, Pagos y Bitácora tendrán un crecimiento constante, debido a que son el corazón del sistema. Considerando todos estos factores, basaremos el cálculo del porcentaje de crecimiento en las tablas de mayor movimiento y para el requerimiento inicial tomaremos en cuenta todas las tablas. La figura 4.2.4.3 nos muestra la longitud de registro de cada una de las tablas que componen nuestra Base de datos.

Es importante mencionar que SQL Server tiene definido por default un 1MB de espacio primario al generar una nueva Base de datos y un 1KB de crecimiento. Aunque también nos permite definir estos dos parámetros por separado.

La fórmula aplicada para estimar la cantidad de espacio requerido para almacenar los datos de una tabla se detalla a continuación:

- Especifique el número de filas presentes en la tabla:  
$$\text{Número de filas de la tabla} = \text{númFilas}$$
- Número de columnas =  $\text{númCols}$
- Suma de los bytes de todas las columnas de longitud fija =  $\text{datosTñoFijo}$
- Número de columnas de longitud variable =  $\text{númColsVariables}$
- Tamaño máximo de todas las columnas de longitud variable =  $\text{tñoMáxVar}$
- Mapa de bits nulo  
$$(\text{mapaBitsNulo}) = 2 + ((\text{númCols} + 7) / 8)$$
- Tamaño total de las columnas de longitud variable  
$$(\text{datosTñoVar}) = 2 + (\text{númColsVariables} \times 2) + \text{tñoMáxVar}$$
- Tamaño total de la fila  
$$(\text{tñoFila}) = \text{datosTñoFijo} + \text{datosTñoVar} + \text{mapaBitsNulo} + 4.$$
- Número de filas por página  
$$(\text{filasPorPág}) = (8096) / (\text{tñoFila} + 2)$$
- Número de filas libres por página

$$(\text{filasLibresPorPág}) = 8096 \times ((100 - \text{factorRelleno}) / 100) / (\text{tñoFila} + 2)$$

- Número de páginas

$$(\text{númPágs}) = \text{númFilas} / (\text{filasPorPág} - \text{filasLibresPorPág})$$

- Tamaño de la tabla

$$(\text{bytes}) = 8192 \times \text{númPágs}$$

En las tablas 4.2.4.3 y 4.2.4.4 se muestran los cálculos realizados para cada una de las tablas del sistema.

TABLA	numfilas	númCols	datosTñfijo	numcolvariables	tñomaxvar	mapabitnulo
BITACORA	100	6	4	5	256	3
CAT_CONCEPTO	8	2	2	1	30	3
CAT_CONTACTO_IF	17	4	3	3	50	3
CAT_DEPTO	83	2	3	1	55	3
CAT_MOTIVO_RECHAZO	8	2	2	1	30	3
CAT_STATUS	6	2	2	1	30	3
CAT_TIPO_PAGO	6	2	2	1	30	3
MENUS	38	3	2	2	50	3
USUARIO	20	5	5	3	50	3
USUARIOSACCESO	760	2	2	1	8	3
PAGOS	50	12	49	2	8	4
ASOCIAPAGO	45	5	12	2	50	3
LINEAEMPRESARIAL	40	16	34	10	50	4
DOCXCOBRAR	5	6	26	0	0	3
CETES	5	2	12	0	0	3

Figura 4.2.4.3 Espacio Requerido

TABLA	datos	TñoVar	TñoFila	FilasPag	Filas_libres_por_pag	numPag	KBytes
BITACORA	268	279	28	20	13	104	
CAT_CONCEPTO	34	43	179	125	1	8	
CAT_CONTACTO_IF	58	68	115	80	1	8	
CAT_DEPTO	59	69	114	79	3	24	
CAT_MOTIVO_RECHAZO	34	43	179	125	1	8	
CAT_STATUS	34	43	179	125	1	8	
CAT_TIPO_PAGO	34	43	179	125	1	8	
MENUS	56	65	120	84	2	16	
USUARIO	58	70	112	78	1	8	
USUARIOSACCESO	12	21	352	246	8	64	
PAGOS	14	71	110	77	2	16	
ASOCIAPAGO	56	75	105	73	2	16	
LINEAEMPRESARIAL	72	114	69	48	2	16	
DOCXCOBRAR	2	35	218	153	1	8	
CETES	2	21	352	246	1	8	
						320KB	

Figura 4.2.4.4 Espacio Requerido (continuación)

Tomando en cuenta todos los datos proporcionados por la figuras 4.2.4.3 y 4.2.4.4, tenemos un requerimiento de espacio inicial de 320KB de datos, más espacio de índices y tablas propias del sistema, y un crecimiento del 10%.

#### 4.2.5 Seguridad Adicional

Para lograr un mejor control en el manejo de datos dentro del sistema se ha implantado un mecanismo adicional de seguridad. Este mecanismo tiene como tarea el registrar en una tabla llamada Bitácora, todos los movimientos realizados por los usuarios dentro del sistema. La Bitácora contiene una estructura sencilla que nos permite conocer: el usuario que realizó el cambio, la tabla que fue alterada, los datos modificados, la fecha y hora del cambio.

El mecanismo para generar la bitácora está basado en los eventos TRIGGERS o desencadenadores, los cuales son ejecutados al realizarse una acción específica como DELETE, INSERT o UPDATE a una tabla. Todas las tablas sensitivas del sistema cuentan con 3 TRIGGERS asociados: el primero que es lanzado al momento ejecutar un DELETE a un de registro; el segundo, es lanzado al momento de ocurrir un INSERT

y por último, el tercero cuando se realiza un UPDATE. Cada uno de estos eventos registra en la tabla Bitácora la acción efectuada, la tabla que fue afectada, y el usuario entre otros.

```
CREATE TRIGGER [NuevoCatContactoIf]
ON [dbo].[CAT_CONTACTO_IF]
FOR INSERT AS
Insert Into Bitacora
Select  GetDate(),
        CONVERT(char(20), SESSION_USER),
        'CAT_CONTACTO_IF',
        'INSERT',
        ins.Cve_If + ' ' + ins. Nombre_If + ' '
        + ins.Contacto + ' ' + ins.Telefono,
        NULL
FROM
inserted ins
```

```
CREATE TRIGGER [BorraCatContactoIf]
ON [dbo].[CAT_CONTACTO_IF]
FOR DELETE AS
Insert Into Bitacora
Select  GetDate(),
        CONVERT(char(20),
        SESSION_USER),
        'CAT_CONTACTO_IF',
        'DELETE',
        NULL,
        del.Cve_If + ' ' + del. Nombre_If + ' ' +
        del.Contacto + ' ' + del.Telefono
FROM
deleted del
```

**Figura 4.2.5.1 Desencadenadores de INSERT y DELETE para el catálogo de Intermediarios Financieros**

El diseño de los desencadenadores se realizó directamente desde la consola Enterprise Management. La figura 4.2.5.1 nos muestra el código de los desencadenadores de INSERT y DELETE.

Con la implantación de los desencadenadores, se puede conocer de forma precisa los cambios realizados a cualquier tabla sensitiva, el usuario que realizó el cambio y los datos afectados. La figura 4.2.5.2 muestra algunos registros generados en la bitácora; en esta figura podemos observar que el Usuario FER076 otorgó algunos permisos al usuario ROA, con lo cual se insertaron los registros de INSERT en la tabla de UsuariosAcceso. También podemos observar que el mismo usuario FER076 se quitó los accesos sobre el menú 6 y se generaron los registros de baja de permisos en la misma tabla.

De esta forma se puede conocer en cualquier momento todos los movimientos realizados en el sistema.

FECHA	CVE USUARIO	TABLA	OPERACIÓN	DATOS NEW	DATOS OLD
29/08/2002 06:29:50 p.m.	FER076	USUARIOSACCESO	INSERT	2 ROA	<NULL>
29/08/2002 06:29:51 p.m.	FER076	USUARIOSACCESO	INSERT	3 ROA	<NULL>
29/08/2002 06:29:54 p.m.	FER076	USUARIOSACCESO	INSERT	5 ROA	<NULL>
29/08/2002 06:29:55 p.m.	FER076	USUARIOSACCESO	INSERT	11 ROA	<NULL>
29/08/2002 06:29:57 p.m.	FER076	USUARIOSACCESO	INSERT	7 ROA	<NULL>
29/08/2002 06:29:58 p.m.	FER076	USUARIOSACCESO	INSERT	14 ROA	<NULL>
29/08/2002 06:30:00 p.m.	FER076	USUARIOSACCESO	INSERT	18 ROA	<NULL>
29/08/2002 06:30:01 p.m.	FER076	USUARIOSACCESO	INSERT	16 ROA	<NULL>
29/08/2002 06:30:02 p.m.	FER076	USUARIOSACCESO	INSERT	22 ROA	<NULL>
29/08/2002 06:30:03 p.m.	FER076	USUARIOSACCESO	INSERT	28 ROA	<NULL>
29/08/2002 06:30:04 p.m.	FER076	USUARIOSACCESO	INSERT	25 ROA	<NULL>
29/08/2002 06:39:35 p.m.	FER076	AsociaPago	INSERT	2 28 Ago 29 2002 6:39PM ROA A	<NULL>
29/08/2002 06:42:22 p.m.	FER076	CAT_DEPTO	UPDATE	5 X	<NULL>
29/08/2002 06:55:15 p.m.	FER076	USUARIOSACCESO	DELETE	<NULL>	6 FER076

Figura 4.2.5.2 Bitácora.

## Usuarios

El modo de conexión de SQL Server que se utilizó fue por autenticación de Windows, es decir que se requiere un solo password para acceder a la Base Datos, mismo que es verificado contra el registrado en el sistema, dentro de la tabla Usuario. Se verifica que el usuario corresponda con el mismo que se firma y también que la contraseña digitada en la pantalla de bienvenida, corresponda al usuario registrado en la tabla.

### **4.3 Diseño y Construcción del Frontend**

- **Diseño**

Una vez realizada la instancia de análisis, la siguiente etapa es la de diseño. El proceso de diseño traduce requisitos en una representación del software que se pueda evaluar por su calidad antes de que comience la codificación o programación.

El diseño es el primer paso en la fase de desarrollo de cualquier sistema. El diseño representa un alto nivel de abstracción, un nivel que se puede seguir hasta requisitos específicos de datos, funcionales y de comportamiento.

El diseño del software (figura 4.3.1) es la primera de tres actividades técnicas (diseño, codificación y pruebas) necesarias para construir y verificar el software. Cada actividad transforma la información de manera que se obtiene un software válido.

En esta etapa de diseño se lleva a cabo un proceso de muchos pasos que se centra en distintos atributos de un programa: estructuras de datos, arquitectura del software, representaciones de interfaz y detalle procedural (algoritmo).

El diseño de datos transforma el modelo de dominio de la información creado durante el análisis en las estructuras de datos necesarias para implementar el software. El diseño arquitectónico define la relación entre los principales elementos estructurales de la aplicación. El diseño de la interfaz describe como se comunica el software consigo mismo, con los sistemas que operan con él y con las áreas usuarias que lo emplean. El diseño procedural transforma elementos estructurales de la arquitectura del programa en una descripción modular de los componentes del software.

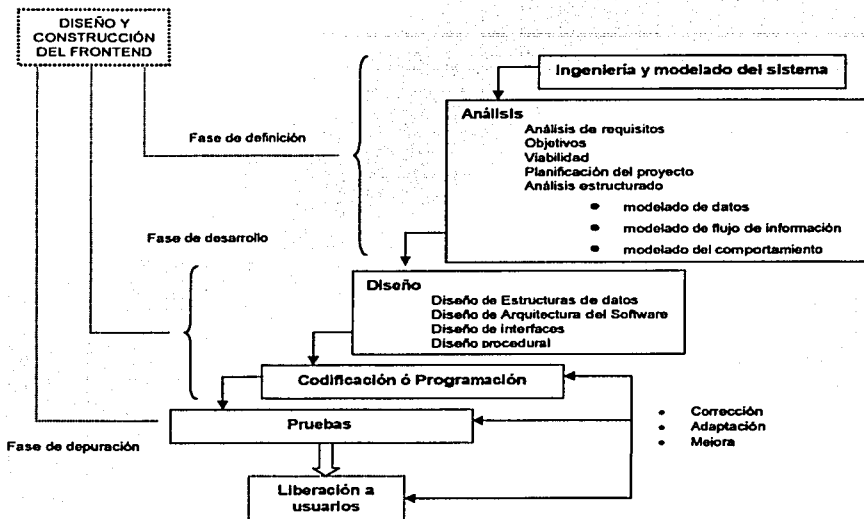


Figura 4.3.1 Fases en el diseño y construcción del Frontend

Para el trabajo de diseño de cualquier aplicación, se deben tener en consideración factores de calidad externos e internos. Los factores de calidad externos son aquellas propiedades que pueden observar los usuarios (velocidad, fiabilidad, utilidad, etc.). Los factores de calidad internos son los que permiten al diseñador la elaboración de alta calidad desde la perspectiva técnica. Para lograr estos factores deben tenerse en claro conceptos básicos como: abstracción, refinamiento, modularidad, arquitectura de software, jerarquía de control ó estructura del programa y estructura de datos.

La abstracción y refinamiento son conceptos complementarios, la abstracción permite especificar datos y procedimientos, el refinamiento ayuda a revelar detalles a medida que progresa el diseño. De igual forma, la arquitectura del software conlleva modularidad; es decir, se divide el software en componentes identificables y tratables por separado, estos componentes se conocen como módulos, los cuales están integrados para satisfacer los requisitos del programa, además, se agrega la estructura

del programa que se encarga de los módulos asociándoles una cierta jerarquía. La estructura de datos es una representación de la relación lógica entre los elementos individuales de datos. Como la estructura de la información afecta invariablemente al diseño final, la estructura de datos es tan importante como la estructura del programa en la representación de la arquitectura del software.

El diseño de datos es la primera de las cuatro actividades de diseño que se llevan a cabo en el desarrollo de aplicaciones. La actividad principal en el diseño de datos es seleccionar representaciones lógicas de objetos de datos identificadas durante la fase de definición y especificación de requisitos. El proceso de selección puede incluir el análisis de estructuras alternativas para determinar el diseño más eficaz, como lo es diseñar una función que será utilizada varias veces dentro del sistema y que con tan sólo invocarla reduzca la codificación. Como ejemplo se puede apreciar la figura 4.3.2, en la que se muestra parte del código de la función *f\_calcula\_dias\_diferencia()*, utilizada para el cálculo de los días que han transcurrido de un documento de pago devuelto. Otra actividad es identificar aquellos módulos del programa que deben operar directamente sobre las estructuras lógicas.

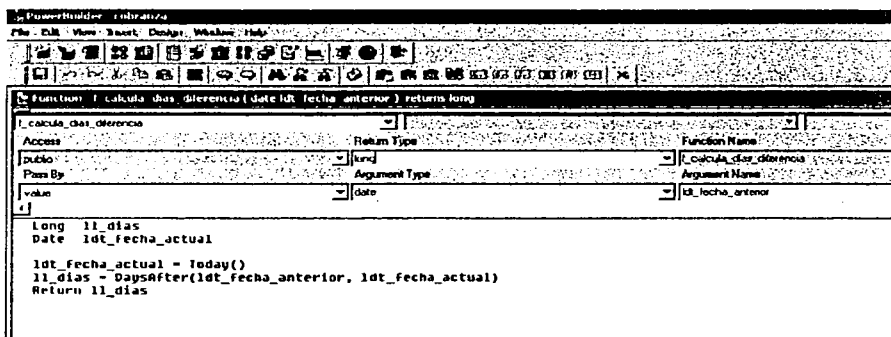


Figura 4.3.2 Función para calcular los días que han transcurrido



Como comentario, unos datos bien diseñados, pueden conducir a una mejor estructura y modularidad de la aplicación, y a una menor complejidad.

Después del diseño de datos, continúa el diseño arquitectónico del software. Su objetivo primario es desarrollar una estructura de programa modular y representar las relaciones de control entre los módulos. Además, el diseño arquitectónico combina la estructura del programa y las estructuras de datos, definiendo interfaces que permiten el flujo de datos a través del programa.

Es importante resaltar que la simplicidad estructural a menudo refleja eficacia. La optimización del diseño arquitectónico debe conseguir el menor número de módulos, pero tantos como sean necesarios; y que la estructura de datos sea menos compleja para satisfacer los requisitos de información.

La herencia nos reduce líneas de código y además provee modularidad al sistema, lo que deriva en sencillez, ya que se vuelve más fácil comprender el flujo de información a través del código. Como podemos observar en las figuras 4.3.3 y 4.3.4, las ventanas heredadas permiten ejecutar un solo código "general" o común entre objetos, y luego ejecutar el código que es particular de cada elemento.

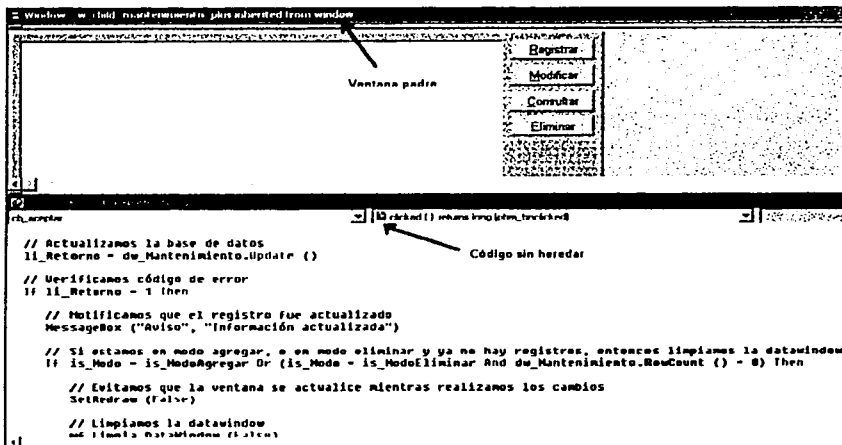


Figura 4.3.3 Ventana papá (código anterior)

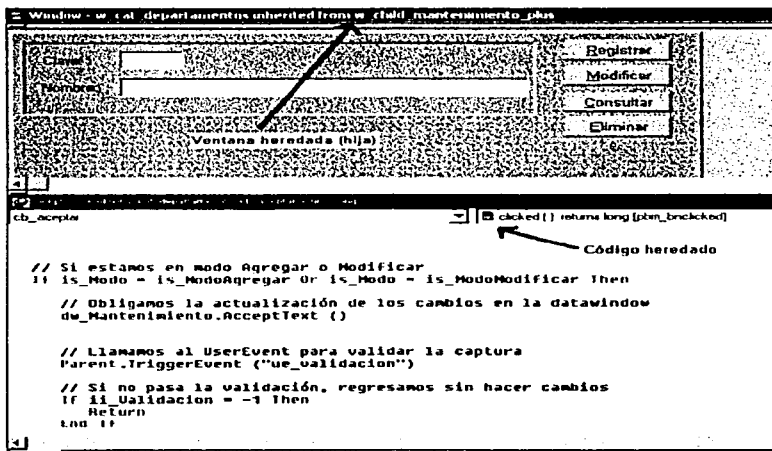


Figura 4.3.4 Ventana heredada o "hija" (código posterior)

El diseño de la interfaz se concentra en tres áreas importantes: diseño de interfaces entre los módulos de software; diseño de interfaces entre el software y otros productores y consumidores no humanos de información; y el diseño de interfaz entre el usuario y la aplicación.

El diseño de interfaces internas del programa depende de los datos que deben fluir entre los módulos y las características del lenguaje de programación en que se va implementar el software, en nuestro caso PowerBuilder versión 7.0 (ver figura 4.3.5).

Para el diseño de la interfaz externa se empieza con una evaluación de cada entidad externa que interviene. Se determinan los requisitos de datos y el control de la entidad para posteriormente diseñar las interfaces apropiadas.

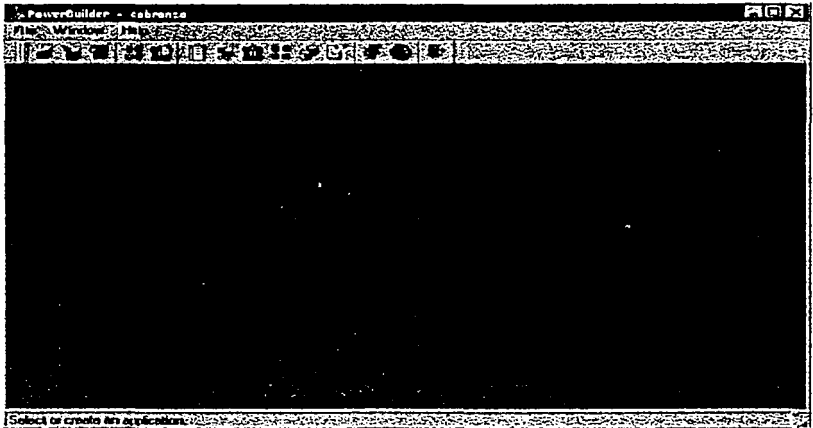


Figura 4.3.5 Ventana principal de desarrollo en PowerBuilder

En nuestro caso, se requiere que el sistema contenga una interfaz a través de menús desde los cuales sea disparado un objeto de tipo ventana (window) en donde el usuario manipulará la información que ingrese o consulte del sistema. El menú que el usuario visualiza se muestra en la figura 4.3.6.

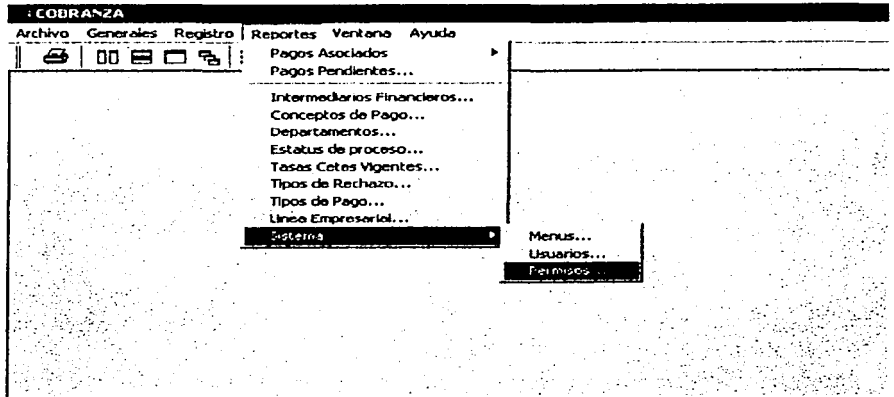


Figura 4.3.6 Menú del Sistema de Cobranza

El diseño procedural se realiza después de los diseños de datos, arquitectónico y de interfaz. Los fundamentos de dicho diseño se sustentan en el uso de un conjunto de construcciones lógicas con las que puede formarse cualquier programa. Como comentario aparte, las construcciones son la secuencia, la condición y la repetición. La secuencia implementa los pasos esenciales en la especificación de un algoritmo, la condición proporciona los medios para seleccionar el procesamiento basándose en alguna ocurrencia lógica y la repetición proporciona los ciclos. Estas tres construcciones son fundamentales para la programación estructurada, que es una importante técnica de diseño.

En la figura 4.3.7 es posible observar un fragmento del código del sistema que incluye estructuras lógicas y de datos utilizadas en la programación.

```
INTEGER ii_existe
gs_ocurrio = trim(cue_usuario.Text)
gs_password = trim(sle_password.Text)
if gs_usuario = "" then
  messagebox("Aviso","No se ha indicado el usuario", stopsign?)
  return
end if
if gs_password = "" then
  messagebox("Aviso","No se ha indicado el password", stopsign?)
  return
end if
select count(*)
into ii_existe
from usuario
where cue_usuario = :gs_usuario
and password = :gs_password
if SQLCA.SQLDBCode <> 0 then
  messagebox("Aviso","Ocurrió un error en el select", stopsign?)
  return
end if
```

Figura 4.3.7 Código fuente con estructuras lógicas

El diseño procedural se lleva a cabo con una serie de herramientas útiles para este fin: diagramas de flujo (vistos en el capítulo 4.1.2), lenguaje de diseño de programas o pseudocódigo, notación tabular de diseño, etc.

La importancia del diseño del software se puede decir con una sola palabra: calidad. El diseño es el lugar donde se fomenta la calidad en el desarrollo del software. El software es la única manera de traducir con precisión los requisitos del usuario en una aplicación.

- **Codificación, Programación o Construcción del Frontend**

El diseño se debe traducir en una forma legible por la máquina. El paso de generación ó codificación de código lleva a cabo esta tarea. Si se lleva a cabo el diseño de una forma detallada, la generación de código se realiza mecánicamente.

La utilización de técnicas de cuarta generación (como es el caso de PowerBuilder) permite estructurar el software a fin de volver a utilizar componentes de programas ya existentes cuando es posible (desarrollo de pantallas padre, que permiten el desarrollo

de pantallas hijo con características heredadas por parte del padre), lo que simplifica la programación y disminuye la redundancia en el código.

De esta forma, al combinar todos los elementos antes mencionados podemos viajar a través del diseño del sistema y conocer el flujo de información como se describe a continuación:

- **Conexión a la Base de Datos, validación de accesos y permisos**

Una aplicación de PowerBuilder es la primera herramienta necesaria para diseñar un sistema. A través de ella es posible realizar la conexión con la Base de Datos (ver figura 4.3.8) y autenticar los accesos (figura 4.3.9) para validar si su intrusión es correcta o no. En caso aprobatorio, el usuario tendrá la posibilidad de ingresar al sistema y tener habilitados los accesos a las opciones del mismo, según los permisos otorgados, previa autorización. De esta forma, los menús sin autorización, aparecerán inhibidos, como se puede apreciar en la figura 4.3.10.

```
sqlca.dbparm = "ConnectString='DSN=MSQL"
SQLCA.database = "Proyecto tesis"
SQLCA.userid = "Administrador"
SQLCA.dbpass = "azkaban"
SQLCA.servername = "LASA 0000000000"
// Profile PROYECTO
SQLCA.DBMS = "ODBC"
SQLCA.lock = "no"
SQLCA.AutoCommit = False
SQLCA.DBParm = "ConnectString='DSN=MSQL;UID=NIEVES;PWD=TESIS',CommitOnDisconnect='No',ConnectOption="

Connect;

If SQLCA.SQLCode <> 0 then
  MessageBox ('Error en la conexión','imposible conectarse a la Base de Datos ~n~n' + &
    SQLCA.SQLErrText)
  HALT
End If
```

Figura 4.3.8 Conexión a la Base de Datos



Figura 4.3.9 Validación de la clave de usuario

▪ Menú principal

El menú contiene todos los objetos a través de los cuales el usuario ingresará a capturar, modificar, consultar o reportar información del sistema. Tendrá la facilidad de disparar un objeto de tipo ventana, a través del cual se realizará la interfaz del sistema con el usuario. Las ventanas contienen todos los códigos necesarios para manipular la información deseada así como ayudas que indican al usuario su estado y utilidad.

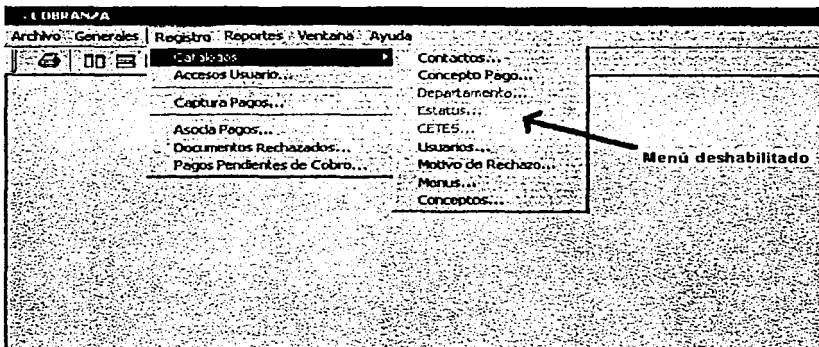


Figura 4.3.10 Menús deshabilitados

- **Ventanas de catálogo**

Las ventanas de catálogos (figura 4.3.11) son objetos que dan mantenimiento (altas, bajas, cambios y consultas) a las tablas “comunes” dentro de toda la aplicación, como pueden ser estatus, conceptos de pago, departamentos, y que son una gran herramienta en la reducción de errores de captura, ya que un usuario sin acceso autorizado, no puede modificar su contenido.

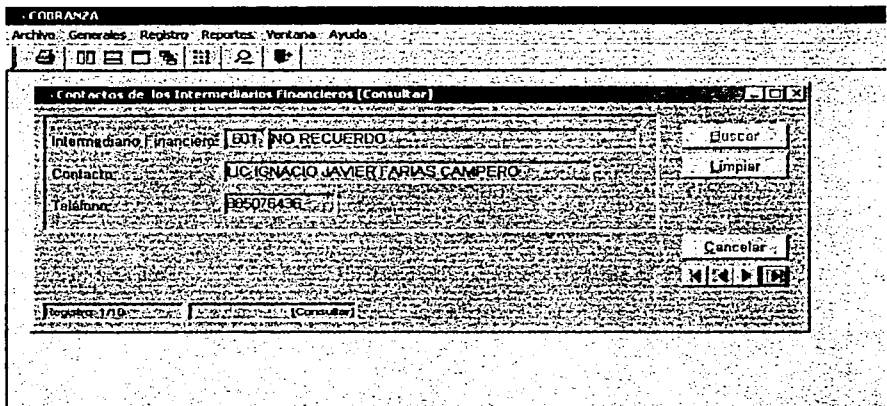


Figura 4.3.11 Catálogo de contactos con Intermediarios Financieros

- **Código (palabras reservadas, estructuras lógicas y validaciones)**

Dentro de los objetos ventana han sido incluidas todas las validaciones necesarias a fin de evitar la captura de datos incorrectos. Se validan los tipos de datos: para no aceptar valores nulos, un formato de fecha incorrecto, un caracter en lugar de un dato numérico, se han puesto filtros en la captura de información, para forzar el ingreso de letras mayúsculas, cuando es necesario. A través del uso de DataWindowsDropDown se evita al usuario la captura de información registrada previamente en los catálogos,



para evitar redundancia en la información y reducir errores humanos de captura. En la figura 4.3.12 se muestra la validación de algunos tipos de datos.

```
w_cal_departamentos | uo_validation | unsigned long wparam, long l
//=====
// Evento      : UE_VALIDACION
// Descripción : Validar datos capturados
//
// Fecha      Autor      Tarea
//-----
// JULIO 2002 PROYECTO TESIS Validar los tipos de datos.
//=====
String ls_nombre, ls_clave
ii_Validation = 1
ls_nombre = de_mantenimiento.GetItemString(1, "nombre_depto")
ls_clave = de_mantenimiento.GetItemString(1, "cve_depto")
IF ISNULL(ls_clave) OR Trim(ls_clave) = "" THEN
ii_Validation = -1
MessageBox("Aviso", "Se debe capturar la clave del Departamento.")
RETURN
END IF
IF ISNULL(ls_nombre) OR Trim(ls_nombre) = "" THEN
ii_Validation = -1
MessageBox("Aviso", "Se debe capturar el nombre de Departamento.")
RETURN
END IF
```

Figura 4.3.12 Código de validación de tipos de datos

Otro aspecto importante en la codificación del sistema son los comentarios escritos dentro del código, a fin de indicar el evento, la persona, la fecha, que programó y modificó el código de ese objeto. Esto ayuda a los programadores en futuras modificaciones y al mantenimiento de los programas en caso de actualizaciones de código.

#### 4.4 Pruebas y liberación del Sistema

Para garantizar la calidad del sistema han de ejecutarse una serie de pruebas que permitan evaluar la capacidad del mismo para procesar la información de manera eficiente, correcta y consistente. Por lo tanto, el objetivo de las pruebas es la ejecución del programa con la intención de descubrir errores; además, deben entrar en juego elementos que garanticen que las pruebas funcionen adecuadamente, con la alta probabilidad de encontrar un error. A su vez, las pruebas nos permiten:

- Asegurar la obtención y formalización de los requerimientos del usuario y verificar que son entendidos de una manera completa, correcta y consistente.
- Verificar los requerimientos funcionales, así como los estructurales y establecerlos como fundamento para realizar las pruebas del sistema.
- Detectar y registrar defectos asociados a los requerimientos establecidos.
- Establecer la documentación de reportes para las pruebas realizadas.

Las pruebas pueden clasificarse en:

##### Unitarias

Se realizan sobre un programa o módulo con la intención de encontrar problemas funcionales en la lógica, así como problemas técnicos en el código, esta prueba centra el proceso de verificación en la menor unidad del diseño del software.

## **Integración**

Este tipo de pruebas se realizan a un grupo de programas para asegurar que los datos y controles sean pasados adecuadamente entre controles. La prueba de integración es una técnica sistemática para construir la estructura del programa mientras que, al mismo tiempo, se llevan a cabo pruebas para detectar errores asociados a la interacción. El objetivo es tomar los módulos probados en unidad y construir una estructura que esté de acuerdo con lo que dicta el diseño.

## **Regresión**

Esta categoría de pruebas selectivas se usan para detectar fallas que pudieran presentarse en algún módulo como consecuencia de modificaciones realizadas; además, permiten verificar que estas modificaciones no afecten negativamente el desempeño del sistema y que se siga cumpliendo con los requerimientos planteados.

## **Volumen**

Se realizan para verificar el funcionamiento adecuado y eficiente de una aplicación bajo condiciones extremas de operación.

## **Caja Negra**

La prueba de la caja negra se realiza con base a los requerimientos sin conocimiento sobre cómo fue construido el sistema y usualmente dirigidas a los datos. Esta prueba intenta encontrar errores de las siguientes características:

- Funciones incorrectas o ausentes
- Errores de interfaz
- Errores en estructura de datos o en accesos a bases de datos externas

- Errores de rendimiento.
- Errores de inicialización y de terminación.

### **Caja blanca**

Estas pruebas se basan en el conocimiento sobre la lógica y estructura interna, usualmente dirigidas a la lógica.

### **Aceptación del usuario.**

Realizadas por el usuario final, para asegurar que el sistema satisfaga las necesidades requeridas.

### **Estáticas**

Revisión y validación de los documentos generados en las distintas fases de la vida de un proyecto.

### **Funcionales**

Validar los requerimientos de la organización (lo que se supone que el sistema debe realizar).

### **Estructurales**

En este tipo de pruebas se valida la arquitectura del sistema confirmando que todas sus partes funcionen de manera armónica y que la tecnología esté siendo utilizada apropiadamente. Se refieren a las características técnicas, como su comportamiento con grandes volúmenes de información, tiempos de respuesta, etc.

## Flujo de información en la prueba

El flujo para la prueba sigue el esquema descrito en la figura 4.4.1 y, como se puede observar, se proporcionan dos clases de entradas al proceso de pruebas:

- Configuración del sistema del software que incluye la especificación del diseño y el código fuente.
- Configuración de pruebas en el que se incluye un plan y procedimiento de prueba.

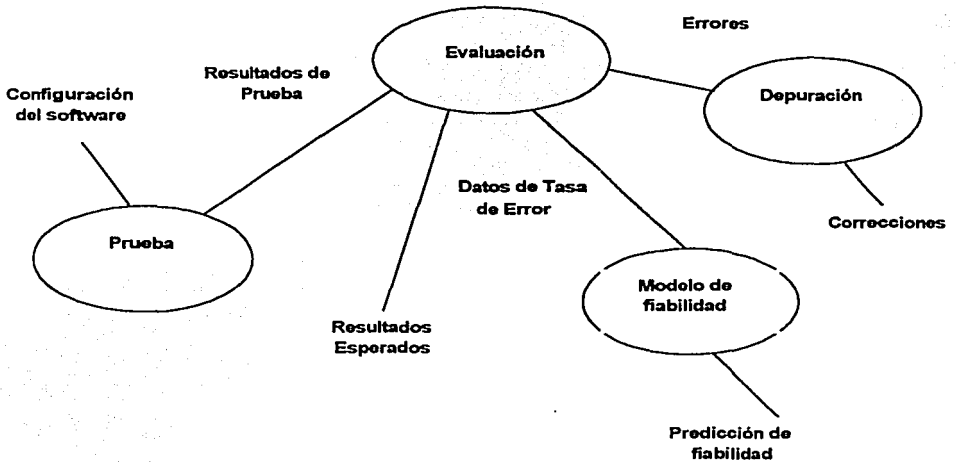


Figura 4.4.1 Flujo de Información en la Prueba

## **Aplicación de pruebas al sistema**

Es importante señalar que la Empresa Hipotecaria cuenta con una metodología de pruebas la cual indica que una aplicación se comprueba con cualquiera de una de estas dos formas: conociendo la función específica para la que fue diseñada, se llevan a cabo pruebas que demuestren que cada función es completamente operativa y al mismo tiempo se buscan errores; y, conociendo el funcionamiento del sistema, se desarrollan pruebas que aseguran que todos sus elementos encajan, es decir, que la operación interna se ajusta a las especificaciones y que todos los componentes internos se han comprobado de forma adecuada.

La primera forma de comprobación se refiere a las pruebas que se llevan a cabo sobre la interfaz del software. O sea, las pruebas que pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce un resultado correcto, y también, que la integridad de la información externa se mantiene.

La segunda forma, se basa en un minucioso examen de los detalles procedurales. Se comprueban los caminos lógicos del software proponiendo casos de pruebas que ejerciten conjuntos específicos de condiciones y/o ciclos.

Considerando lo anterior las pruebas que se realizaron al Sistema de Automatización del Proceso de Cobranza de una Empresa Hipotecaria fueron las siguientes:

### **Pruebas de caja blanca**

Este tipo de pruebas son las primeras que se realizan cuando se desarrolla un sistema, debido a que el programador conoce la lógica del sistema y conforme va realizando cualquier adecuación o desarrollo, se corren pruebas para verificar la información, en muchas ocasiones, estas pruebas se realizan con ayuda de un usuario responsable del módulo o del sistema en general.

Para ejemplificar este tipo de pruebas, se tiene la opción **Captura Pagos** del submenú **Registro**, ver figura 4.4.2, en la cual el usuario debe revisar que se encuentren disponible todos los campos que necesita capturar, además validar que la información cargada previamente dentro del sistema sea la correcta, (por ejemplo los datos del intermediario financiero, monto del pago), tales datos deberán ser cotejados contra los cheques recibidos así como con el catálogo de la empresa.

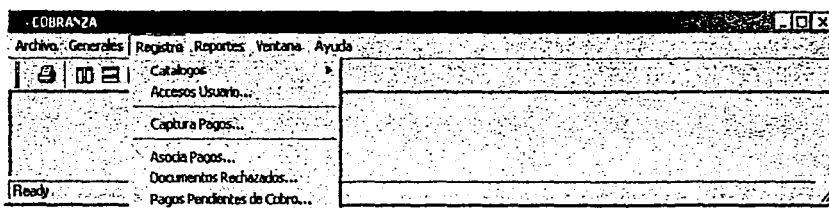


Figura 4.4.2 Pantalla del submenú de registro.

De esta forma, el usuario acceso al módulo de captura de pagos y encontró todos los campos necesarios para desarrollar su trabajo, figura 4.4.3. Para ello el usuario capturó los datos omitiendo intencionalmente algunos campos para probar la respuesta del sistema. En las figuras 4.4.4, 4.4.5, 4.4.6 y 4.4.7 se muestran los mensajes de error que generó el sistema; cabe señalar, que el sistema no permite continuar hasta que el usuario introduzca todos los datos.

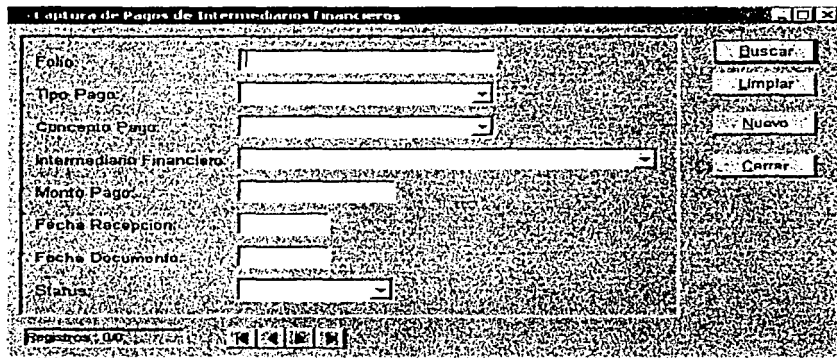


Figura 4.4.3 Pantalla del módulo captura de pagos

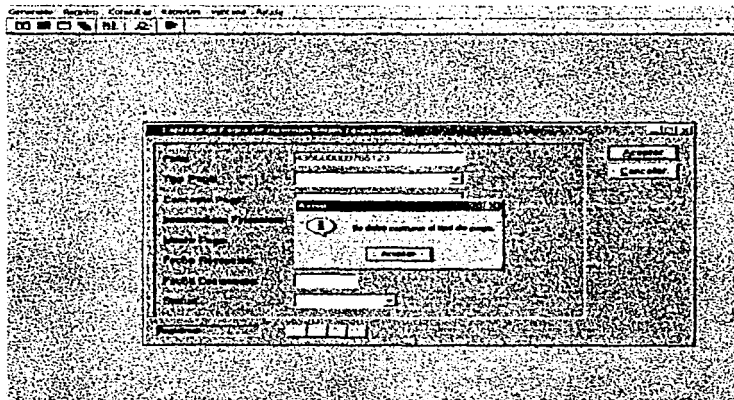


Figura 4.4.4 Error al omitir el tipo de pago.



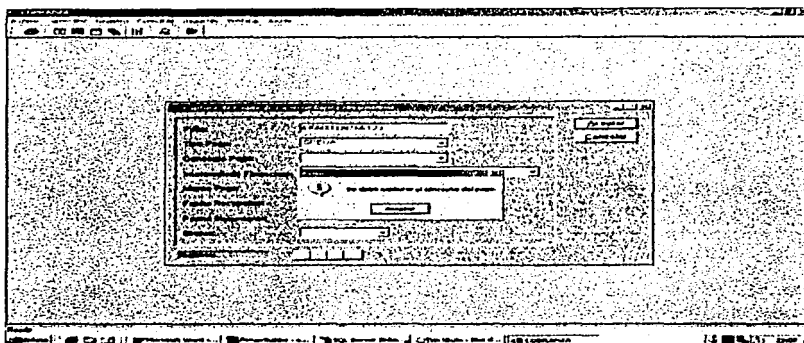


Figura 4.4.5 Ventana de error cuando se omite el concepto de pago.

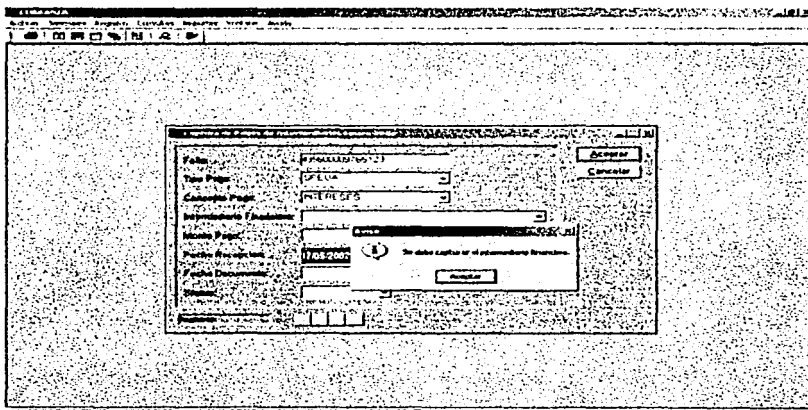


Figura 4.4.6 Error de campo de intermediario financiero incompleto.

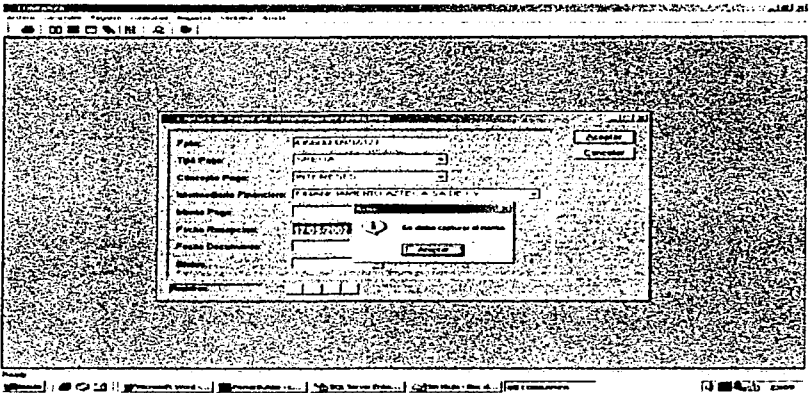


Figura 4.4.7 Mensaje de error generado al no introducir el monto del pago.

Después de realizar estas pruebas se encontraron los siguientes errores:

- Al capturar un nuevo pago con un folio ya registrado previamente, el sistema no indicaba el tipo de error que se generaba, lo cual causaba un poco de confusión al usuario final.

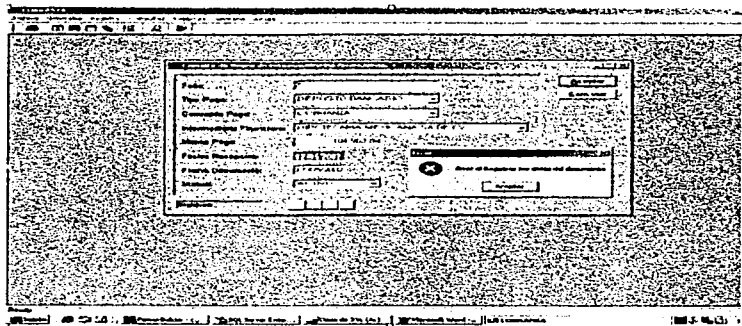


Figura 4.4.8 Ventana de error inconclusa

Posteriormente, se depuró el sistema y se sometió a la misma prueba y en las mismas condiciones iniciales; en esta fase se cumplió con las observaciones hechas por el cliente sin ningún error. En las figuras 4.4.9 y 4.4.10 se observan las ventanas con las modificaciones sugeridas por el usuario, se agregó un aviso del folio duplicado y en seguida envía mensaje de error al capturar los datos.

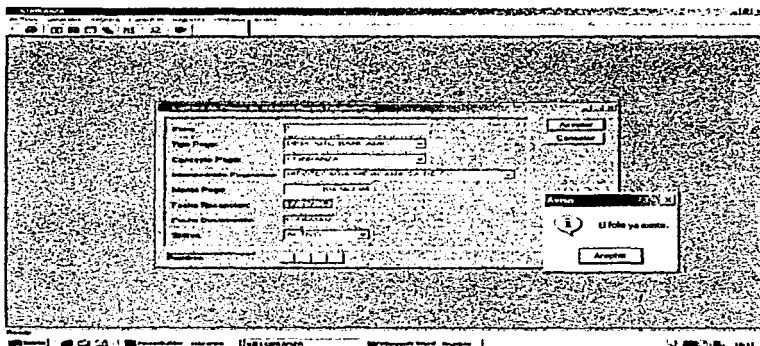


Figura 4.4.9 Mensaje de error especificando el origen de la falla

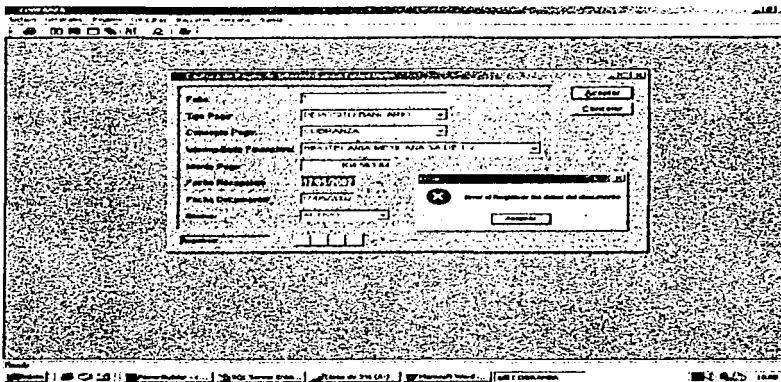


Figura 4.4.10 Mensaje de error al capturar un folio duplicado

### Prueba de caja negra.

Considerando que el sistema está compuesto de módulos específicos para cada actividad se procedió a probar cada módulo con el usuario correspondiente, estas pruebas se realizaron para verificar la correcta captura de pagos, permisos de usuarios, entre otros. A continuación se muestra los resultados del módulo de captura.

### Módulo de Captura

En la figura 4.4.11 se muestra gráficamente la prueba de caja negra para el módulo de captura, la figura 4.4.12 y 4.4.13 la entrada y la figura 4.4.14 la salida.

Entrada: Monto del pago (cheque). El cliente captura la información en la ventana de pagos.

Salida: Confirmación de que el monto del cheque coincide con el monto esperado.

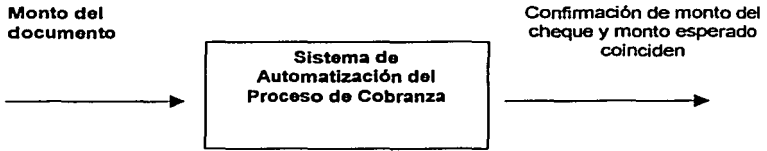


Figura 4.4.11 Prueba de Caja Negra.

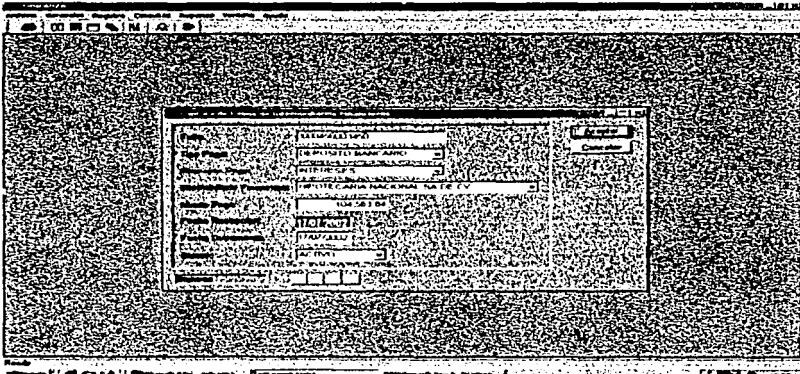


Figura 4.4.12 Registro del pago

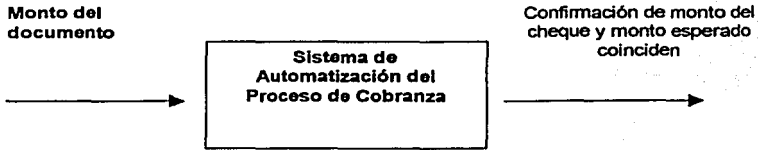


Figura 4.4.11 Prueba de Caja Negra.

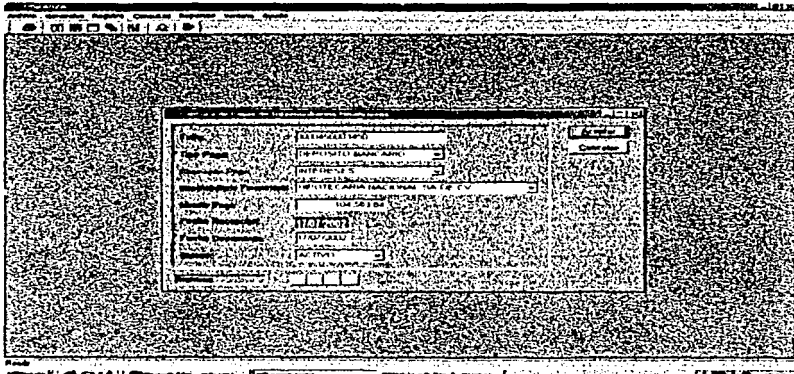


Figura 4.4.12 Registro del pago

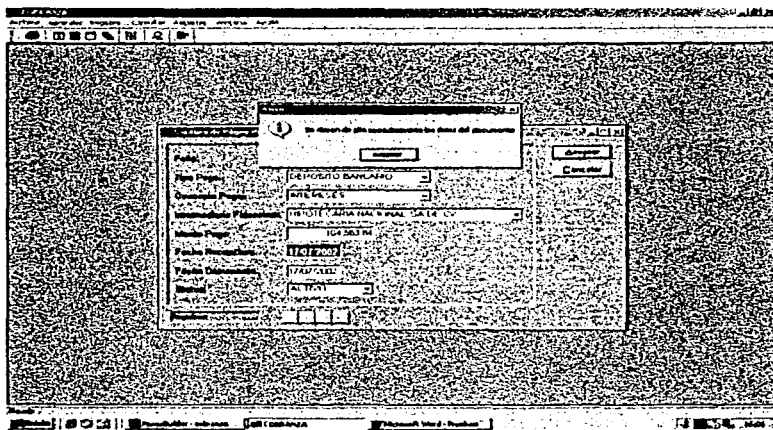
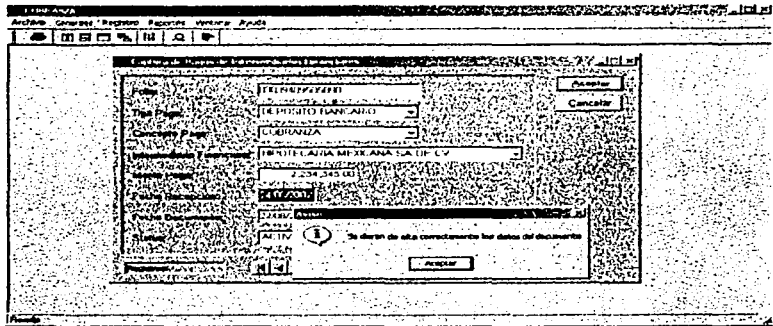


Figura 4.4.13 Mensaje de captura del pago exitosa

CV	CATEGORIA	FECHA	MONTOS
1	COBRO DE CV INTER	23/04/2002	10,000.00
2	APLICACIÓN DE CV	24/04/2002	15,000.00
3	COBRO DE CV INTER	25/04/2002	1,000.00
4	COBRO DE CV INTER	27/04/2002	35,000.00
5	DEPOSITO CON DOCUMENTOS	28/04/2002	1,000.00
6	DEPOSITO CON DOCUMENTOS	29/04/2002	1,000.00
7	DEPOSITO CON DOCUMENTOS	30/04/2002	1,000.00

Figura 4.4.14 Salida, Confirmación de coincidencia de montos.

### Pruebas de integridad

La integridad contempla varios aspectos:

- **Integridad semántica:** Cuando existe validación de la naturaleza de los datos a través de programas o el manejador de bases de datos.
- **Integridad de entidades:** No se debe aceptar nulos en la llave primaria.
- **Integridad Referencial:** Las llaves foráneas corresponden a la llave primaria.

Dichas pruebas también se aplicaron al sistema para validar las reglas establecidas del negocio.

Por ejemplo, al tratar de introducir un elemento nulo en la llave primaria, el sistema manda un error y se le tiene que teclear otro valor diferente ya establecido, obsérvese la figura 4.4.15.



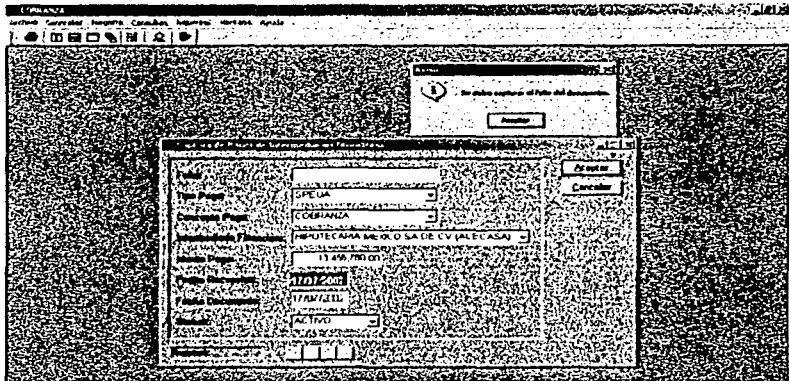


Figura 4.4.16 Error al omitir el campo de la llave primaria del módulo de captura de pagos

Después de terminadas las pruebas y la depuración del sistema se procedió a entrenar a los usuarios para continuar con la liberación de éste. Sin embargo, una vez liberada la aplicación ya probada a las áreas usuarias se pueden llegar a presentar algunos problemas. Dentro de los problemas más comunes están: primero, en la etapa de análisis, es común que el usuario difícilmente exponga explícitamente todos los requisitos; segundo, también por parte del usuario emergen detalles que pertenecen a la etapa de diseño: el tiempo de respuesta del sistema, las facilidades de ayuda al usuario, la manipulación de información de errores, etc.; ó tercero, debido a la evolución de su entorno, la aplicación liberada debe sufrir cambios dado que el usuario requiere mejoras funcionales o de rendimiento.

Por lo anterior se efectuaron las pruebas alfa y beta antes de liberar completamente el sistema.

La prueba alfa es conducida por un usuario en el lugar de desarrollo (entorno controlado). Se usa el software de forma natural, con el encargado de desarrollo "mirando" al usuario y registrando errores y problemas de uso.



**Fig. 4.4.16 Desarrollo de prueba alfa**

La prueba beta se lleva a cabo en uno o más lugares del cliente y es operado por los usuarios finales del software. A diferencia de la prueba alfa, el encargado del desarrollo no esta presente. Así la prueba beta es una aplicación "en vivo" del software en un entorno que no puede ser controlado por el equipo de desarrollo. El cliente registra todos los problemas (reales o imaginarios) que encuentra durante la prueba beta e informa a intervalos regulares al equipo de desarrollo. Como resultado de los problemas anotados durante la prueba beta, el equipo de desarrollo del software lleva a cabo modificaciones y así prepara una versión del sistema para toda la base de usuarios.

Después del desarrollo de estas pruebas sólo se modificaron algunas ventanas de error para facilitar al usuario la identificación del mismo y su posterior corrección. De igual forma se agregó un aviso al eliminar un registro del catálogo de departamentos, pues se asumía de antemano que el cliente estaba seguro de realizarlo, véase la figura 4.4.17. Este cambio se realizó a todos los catálogos.

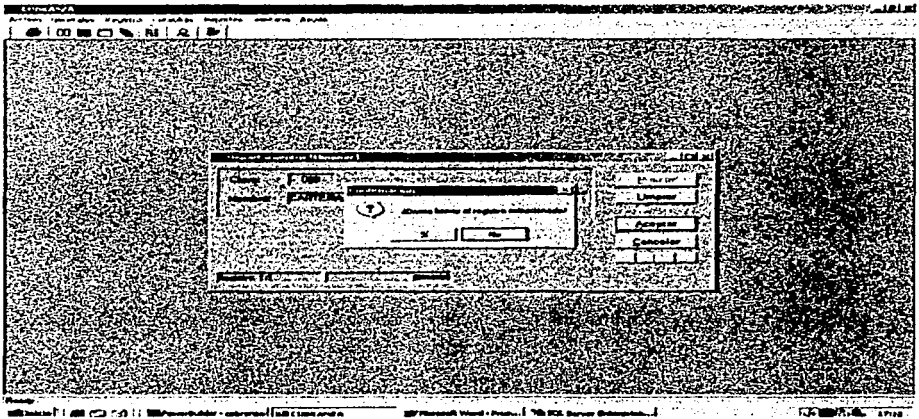


Figura 4.4.17 Mensaje de confirmación para eliminar registro

## 4.5 Generación de Reportes

Para la generación de reportes utilizamos una herramienta propia de PowerBuilder mencionada en el capítulo 3.6.8.1, conocida como datawindow, la cual involucra funciones para presentar la información (color de fuente, opciones de impresión, múltiples objetos de operaciones aritméticas y lógicas, inserción de objetos) y que permite interaccionar directamente con la base de datos a través de sentencias SQL, siendo posible hacer uso de estructuras lógicas (cláusulas WHERE, FOR, IF, entre otros) para obtener los datos deseados.

Para crear una datawindow se debe acceder al menú Archivo \ Nuevo, en la pestaña DataWindow, mostrado en la figura 4.5.1.

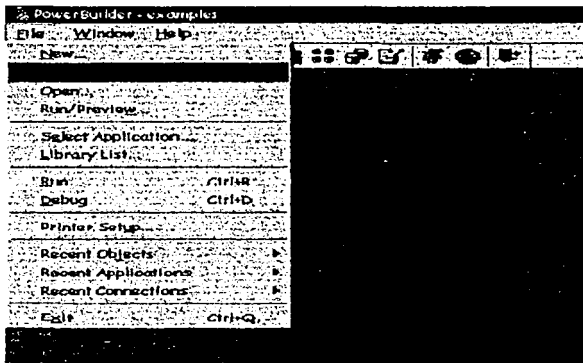


Figura 4.5.1 Menú de acceso a la generación de datawindows

Se desplegará una ventana con diferentes opciones, según la manera en que se desea desplegar la información (figura 4.5.2).

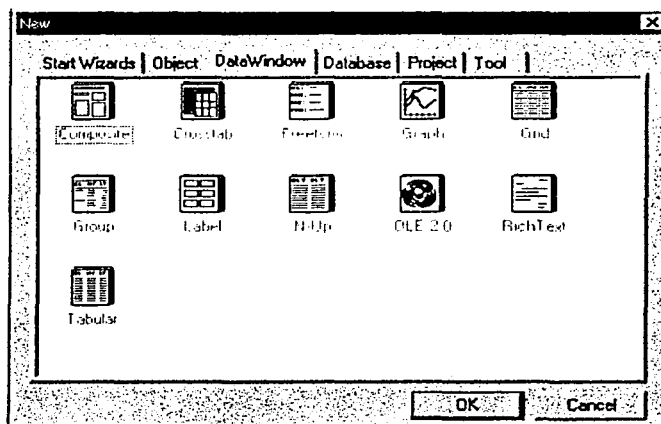


Figura 4.5.2 Diferentes formas de datawindows

- Composite:** Genera un reporte a partir de otra datawindow (es decir, de forma compuesta), en la cual existe la opción para agregar parámetros de consulta.
- Crosstab:** Permite un cruzamiento de datos, a fin de obtener una estructura de tipo "OR".
- Freeform:** Despliega los datos sin un orden específico, para tener la facilidad de acomodarlos como el programador lo desee.
- Graph:** Genera un reporte de tipo gráfica.
- Grid:** Es del tipo tabular (por columnas), pero los anchos de columnas son continuos.
- Group:** Agrupa los datos según se desee, desde el diseño.
- Label:** Despliega la información con apariencia de etiqueta.
- N-up:** Despliega la información en varias columnas (por bloques de columnas de tablas), para ahorrar espacio, cuando las columnas de las tablas son pocas.

- OLE:** Permite utilizar objetos externos a PowerBuilder (archivos de Excel, Word, PowerPoint, Paint, entre otros) dentro de los reportes, para desplegar información que contenga vínculos con otras aplicaciones.
- Rich Text:** Permite editar o diseñar el reporte utilizando texto enriquecido (fuentes true type, Post script, colores, formas, dibujos y más).
- Tabular:** Muestra los datos por columnas, permitiendo su libre acomodo.

Posteriormente se debe elegir el tipo de query (consulta) que se desea utilizar para generar el código de la datawindow (ver figura 4.5.3). Este puede ser por medio de una selección rápida, a través de un procedimiento, de una función externa, entre otras.

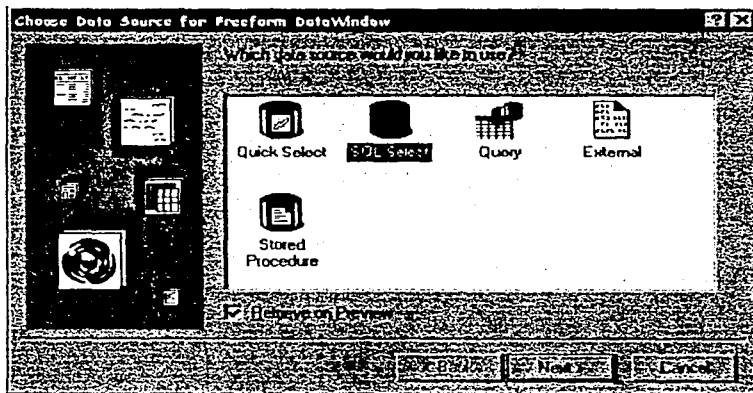


Figura 4.5.3 Tipo de query para realizar la consulta de información

Al indicar una de ellas y dependiendo del tipo de reporte que se haya elegido, aparecerá, de forma gráfica, una lista de todas las tablas disponibles del sistema, a fin de elegir aquella o aquellas sobre las cuales se obtendrán los datos, permitiendo realizar las ligas correspondientes. En esta ventana (figura 4.5.4) se dispone de cuatro pestañas, mismas que pueden ser utilizadas para dar forma y filtrar los datos como sea requerido; la pestaña "Sort" facilita la ordenación de los datos; la pestaña "Where" permite filtrar la información, mostrando las distintas columnas de todas las tablas

utilizadas, los operadores y las sentencias permitidas; la pestaña "Group" agrupa los datos por columnas; la pestaña "Having" filtra los datos según contengan o no cierta información; la pestaña "Compute" genera campos computados para manipular los datos contenidos en las columnas, siendo utilizables dentro del reporte como una columna más; por último, la pestaña "Syntax" muestra la sintaxis generada por todas las opciones anteriores.

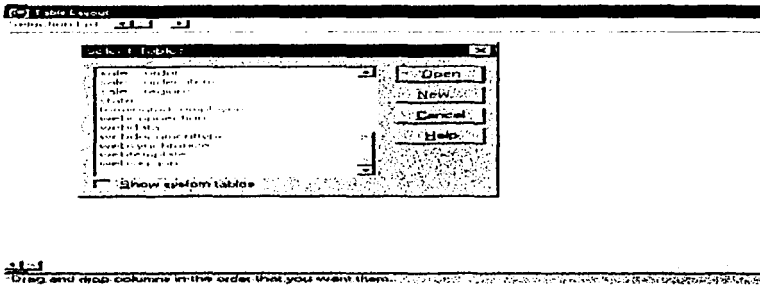


Figura 4.5.4 Lista gráfica de tablas en la creación de datawindows

Aunado a estas opciones, PowerBuilder y el asistente de creación de datawindows, nos dan la oportunidad de salirnos del modo gráfico (figura 4.5.5 y 4.5.6), para editar (utilizando comandos de Transact SQL) nuestra propia sintaxis - en pocas ocasiones es necesario, pues la forma gráfica reduce significativamente el código y facilita su creación -.

Una vez terminada la generación del código, con la opción "Return" regresamos el control al diseño visual del reporte. Aparece una ventana con las opciones de fondo, fuente, forma y color para darle presentación a la datawindow, misma que se puede

observar en la figura 4.5.7. Por último, sólo es necesario oprimir el botón "Finalizar" y podremos observar nuestro reporte.



Figura 4.5.5 Convierte la generación de la datawindow en sintaxis de SQL

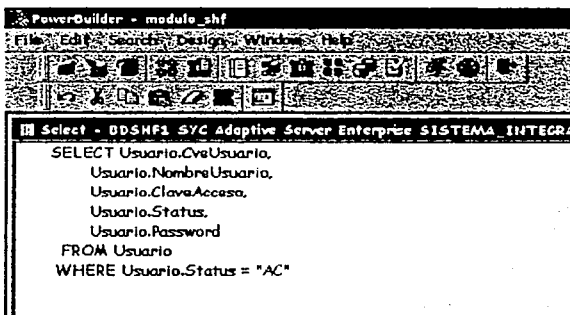


Figura 4.5.6 Sintaxis generada para la consulta de la datawindow

Dentro de la ventana de la datawindow ya generada, podemos modificar las opciones de fuente, color, forma, opciones de impresión e incluso insertar más objetos como fecha, número de páginas, campos con operaciones aritméticas y lógicas. La misma datawindow posee diferentes espacios para desplegar la información (ver figura 4.5.8): el "Header" en el cual se pueden poner los encabezados fijos para todas las páginas



que incluye el reporte, es la primera fila que muestra el reporte; "Detail", es el contenido a detalle de los renglones del reporte; "Summary", para sumarizar los diferentes campos del reporte, es el último registro del contenido del reporte; "Footer", para insertar un pie de página, si se desea.

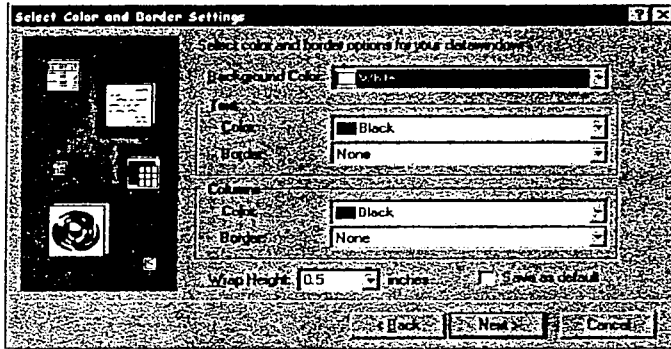


Figura 4.5.7 Opciones de diseño final en una datawindow

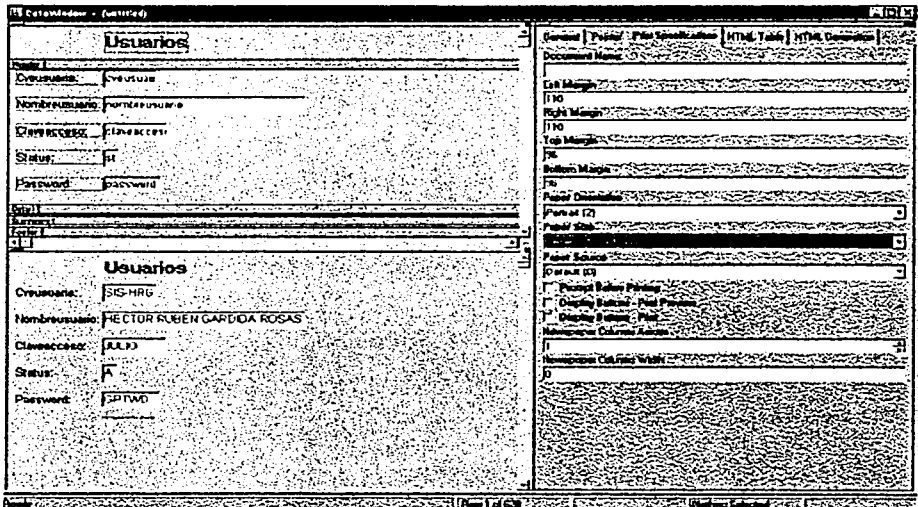


Figura 4.5.8 Opciones de diseño de una datawindow.

Dentro de todos los sistemas la forma de conocer de forma concisa el estado que guardan los datos es a través de los Reportes. Los reportes del sistema deben contener información completa, de fácil interpretación y confiable.

Los reportes generados por Sistema están disponibles a través del menú de Reportes, el cual se muestra en la figura 4.5.9.

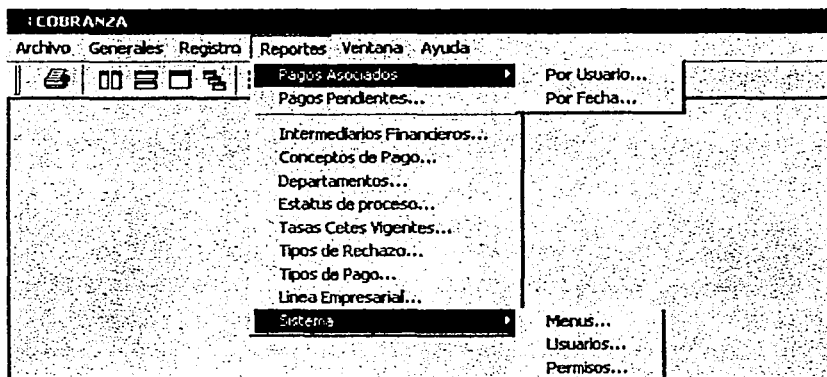


Figura 4.5.9 Menú Reportes

### Estructura general de los reportes

Todos los reportes generados por el sistema contienen:

- Fecha de Generación: Esta fecha se muestra en la parte superior Izquierda.
- Nombre de la empresa: Se muestra en la parte central superior del reporte
- Número de Páginas: Siempre se muestra la pagina corriente, mas el total de páginas que integran el reporte.
- Título del Reporte: Se muestra en la parte central en la segunda línea
- Títulos de Columnas: Siempre se muestran en negritas.
- Detalles del reporte: Se muestra en letra normal.

- **Total de registros o líneas de Detalle del Reporte:** Al final de cada reporte en la esquina inferior izquierda.
- **Monto Total:** Este dato solo se muestra si el reporte en cuestión involucra cantidades monetarias, y siempre al final de cada reporte en la esquina inferior izquierda.
- **Ruta del reporte:** Esta información es para localizar de forma el menú que genera el reporte y se localiza en la parte inferior derecha del reporte.

### Reportes de Pagos Asociados

Esta opción nos permite generar dos reportes sobre los Pagos asociados los cuales pueden apreciarse en la figura 4.5.10 y 4.5.11:

- **Pagos Asociados por Usuario.** Que contiene todos los pagos que fueron asociados por un usuario en particular, el cual se puede escoger mediante un listdropdown que nos muestra todos los usuarios existentes en el sistema. El reporte contiene toda la información necesaria para conocer en forma precisa; el Intermediario Financiero que realizó el pago, el monto, el tipo de pago que fue recibido y el concepto del mismo. Además nos proporciona la fecha en la cual el pago fue asociado y su fecha de recepción.

Reporte de Pagos Asociados por Usuario

Fernando

Fecha 25/08/02

Empresa Hipotecaria, S.A. de C.V.

Página

**Reporte de Pagos Asociados**

FERN DIRECCION GENERAL

Intermediario Financiero	Concepto	Tipo de Pago	Total Pagado	Rt
HIPOTECARIA VANGUARDIA SA DE CV	COBRANZA	DEPOSITO BANCARIO	2,978,356.30	2
FINANCIAMIENTO AZTECA SA DE CV	COBRANZA	DEPOSITO BANCARIO	25,076,528.00	2
NO RECUERDO	COBRANZA	SPEUA	1.00	0
METROFINANCIERA SA DE CV	COBRANZA	DEPOSITO BANCARIO	23,490.00	0
NO RECUERDO	COBRANZA	SPEUA	1.00	2

Configurar Imp... Imp. Predefinir Imprimir Cerrar

Figura 4.5.10 Pagos asociados por usuario

- Pagos Asociados por Fecha: Contiene todos los pagos asociados a una fecha en particular. Las fechas de asociación se escogen mediante un menú Dropdown. Su estructura es similar al reporte de Pagos Asociados por usuario.

Reporte de Pagos Asociados por Usuario

21/07/2002

24/08/2002

17/08/2002

17/08/2002

30/07/2002

17/08/2002

Empresa Hipotecaria, S.A. de C.V.

Página 1 de 3

**Reporte de Pagos Asociados del día 21/07/2002**

Intermediario Financiero	Concepto	Tipo de Pago	Total Pagado	Fecha R
HIPOTECARIA VANGUARDIA SA DE CV	COBRANZA	DEPOSITO BANCARIO	2,978,356.30	20/07
FINANCIAMIENTO AZTECA SA DE CV	COBRANZA	DEPOSITO BANCARIO	25,076,528.00	20/07
PATRIMONIO SA DE CV	COBRANZA	SPEUA	1.00	05/07
Total de Registros:		8		
Total Asociado		20,115,818.32		

Configurar Imp... Imp. Predefinir Imprimir Cerrar

Figura 4.5.11 Pagos asociados por fecha

## Reportes de Pagos Pendientes

Contiene la información de todos los pagos pendientes de Asociarse debido a que fue rechazado el pago. Este reporte contiene información necesaria para realizar el seguimiento con el contacto Financiero y solicitar el pago oportuno de la deuda. La información desplegada en el reporte es: El nombre del Intermediario Financiero que realizó el pago, el tipo y concepto del pago realizado, la fecha de Recepción, El monto que debió ser cubierto, los intereses generados a la fecha, - en este punto es importante mencionar que los intereses son calculados al momento de generar el reporte y de acuerdo al Valor del CETE registrado vigente -. También nos muestra el nuevo monto adeudado, y contacto con el cual se debe llevar el seguimiento y el teléfono del mismo.

La figura 4.5.12 muestra el reporte de pagos pendientes.

Reporte de Pagos Pendientes de Asociación

Fecha 25/08/02

Empresa Hipotecaria, S.A. de C.V. Pag 1

Reporte de Pagos Pendientes

Intermediario	Tipo de Pago	Concepto	Fecha de Recepción	Pagos Monto	Intereses Generados	Ad
HIPOTECARIA VANGUARDIA SA DE CV	DEPOSITO BANCARIO	COBRANZA	20/07/2002	2,978,366.30	32166.24	2
HIPOTECARIA MEXICANA SA DE CV	DEPOSITO BANCARIO	COBRANZA	17/05/2002	104,563.84	3136.92	2
FINANCIAMIENTO AZTECA SA DE CV	DEPOSITO BANCARIO	COBRANZA	16/05/2002	109,993,543.00	3332004.35	110
HIPOTECARIA MEXICANA SA DE CV	DEPOSITO BANCARIO	COBRANZA	24/08/2002	2,234,345.00	670.3	2
FINANCIAMIENTO AZTECA SA DE CV	DEPOSITO BANCARIO	COBRANZA	20/07/2002	25,076,528.00	270826.5	25
HIPOTECARIA VANGUARDIA SA DE CV	DEPOSITO BANCARIO	COBRANZA	16/05/2002	104,563.84	3168.29	2

REPORTES/PAGOS

Configurar Imp... Impr. Prácticas Imprimir Cerrar

Figura 4.5.12 Reporte de pagos pendientes de Asociación

## Reportes de Catálogos

Los reportes de catálogos son un conjunto de reportes que nos muestran la información contenida en los catálogos del sistema. Todos los reportes de catálogos son mostrados en las figuras 4.5.13, 4.5.14, 4.5.15, 4.5.16, 4.5.17, 4.5.18, 4.5.19 y 4.5.20.

Dentro de los reportes de catálogos del sistema tenemos:

- Reporte del Intermediarios Financieros: Que nos muestra el nombre del Intermediario Financiero, el contacto y el teléfono del mismo.
- Reporte de Conceptos de Pago. Con este reporte podemos conocer las distintas claves para cada concepto y la descripción existente para cada una.
- Reporte de Departamentos del sistema: Contiene la clave del departamento y el nombre del mismo.
- Reporte de Estatus de Proceso. Nos muestra la clave del estatus y la descripción establecida para el mismo.
- Reporte de CETES: Este reporte nos muestra las fechas registradas para CETES y el valor que se estableció para la tasa de CETES de esa fecha.
- Reporte de Tipos de Rechazo: En este reporte se muestra el catálogo que contiene los motivos por los cuales puede ser rechazado un documento de pago.
- Reporte de Tipos de Pagos: Se muestra la clave del tipo de pago existente y la descripción del mismo.
- Reporte de Línea Empresarial: Muestra de forma general los pagos existentes en cada Línea Empresarial cargada que no han sido asociados.

Reporte de Intermediarios Financieros

Fecha: 25/08/02 Empresa Hipotecaria, S.A. de C.V. Pág 1 de 2

Institución Financiera	Contacto	Telefono
PATRIMONIO SA DE CV	LIC JULIO ROBLES	005076436
HIPOTECARIA MEXICO SA DE CV (ACECASA)	LIC VICENTE ORTEGA	0907654
FINANCIAMIENTO AZTECA SA DE CV	ING LETICIA REYES	0907654
HIPOTECARIA MEXICANA SA DE CV	ING GERARDO NIETO DE LUNA	0907654
HIPOTECARIA NACIONAL SA DE CV	ING ALEJANDRO PALOMERA	0907654
GENERAL HIPOTECARIA SA DE CV	DR RODRIGO SUAREZ	0907654
HIPOTECARIA COMERCIAL AMERICA SA DE CV	C P VALENTINA GUZMAN	0907654
CREDITO INMOBILIARIO SA DE CV	ACT JORGE RESENDIZ	0907654
METROFINANCIERA SA DE CV	LIC GONZALO VELAZQUEZ	0907654

Total de Registros: 17

Configurar Imp... Imp. Predefinir Imprimir Cerrar

Figura 4.5.13 Reporte del Intermediarios Financieros

Reporte de conceptos de Pago

Fecha: 25/08/02 Empresa Hipotecaria, S.A. de C.V. Pág 1 de 1

Clave	Descripción
1	COBRANZA
2	INTERESES
3	PENALIZACION
4	CUOTA INGRESO
5	REACTIVACION

Total de Registros: 5

Configurar Imp... Imp. Predefinir Imprimir Cerrar

Figura 4.5.14 Reporte de Conceptos de Pago

Reporte de Departamentos

Fecha 25/03/02 Empresa Hipotecaria, S.A. de C.V. Pág 1 de 12

**Reporte de Departamentos**

Clave Departamento	Nombre del Departamento
300	DIRECCION GENERAL
301	SECRETARIO DE CONSEJO
302	DIRECCION DE RELACIONES INSTITUCIONALES Y DE COMUN
303	SUBDIRECCION DE PROMOCION
304	DELEGACION DISTRITO FEDERAL
305	DELEGACION GUADALAJARA
306	DELEGACION CANCUN

Total de Registros: 83

Configurar Imp... Imp. Predefinir Imprimir Cerrar

Figura 4.5.15 Reporte de Departamentos de la empresa

Reporte de Motivos de Rechazo de Pagos Recibidos

Fecha 25/03/02 Empresa Hipotecaria, S.A. de C.V. Pág 1 de 1

**Reporte de Motivos de Rechazo de Pagos Recibidos**

Clave	Descripción
1	CHEQUE SIN FONDOS
2	CUENTA CANCELADA
3	CUENTA DENEGADA
4	Inconsistencias en el Pago

Total de Registros: 4

Configurar Imp... Imp. Predefinir Imprimir Cerrar

Figura 4.5.16 Reporte de Tipos de Rechazo



Fecha Publicada	Valor
15/06/2002	14.25
22/06/2002	13.96
29/06/2002	13.56
14/07/2002	1.20
29/07/2002	10.56
05/08/2002	10.00
10/08/2002	6.50

Figura 4.5.17 Reporte de CETES

Clave	Descripción
1	SPEUA
2	DEPOSITO BANCARIO
3	CHEQUE
4	EFFECTIVO EN CAJA

Total de Registros 4

Figura 4.5.18 Reporte de Tipos de Pagos

Reporte de Pagos de la Línea Empresarial No Asociados a la Fecha

Fecha: 25/08/02 Pag 1 de 2

**Empresa Hipotecaria, S.A. de C.V.**

**Línea Empresarial pendiente de Asociar del día 02/08/2002**

Fecha Estado de Cuenta	Movimiento	Descripción	Monto Abonado	Sucursal	Banco Orige
02/08/2002	0	ABONO POR COBRANZA	7294 6300	3005	000
02/08/2002	0	ABONO POR COBRANZA	24107 7000	3005	000
02/08/2002	0	ABONO POR COBRANZA	4902 5000	520	000
02/08/2002	0	ABONO POR COBRANZA	2660 4500	632	000
02/08/2002	0	ABONO POR COBRANZA	7821 5200	632	000
02/08/2002	0	ABONO POR COBRANZA	17072 7700	602	000
02/08/2002	0	ABONO POR COBRANZA	10666 6600	240	000
02/08/2002	0	ABONO POR COBRANZA	9900 6600	201	000

Configurar Imp...    Impr. Predefinir    Imprimir    Cerrar

Figura 4.6.19 Reporte de Línea Empresarial

Reporte de Estatus

Fecha: 25/08/02 Pag 1 de 1

**Empresa Hipotecaria, S.A. de C.V.**

**Reporte de Estatus de Proceso**

Clave	Descripción
AC	ACTIVO
AS	ASOCIADO
AU	AUDITADO
BA	BAJA
CA	CANCELADO
PE	PENDIENTE
TE	TERMINADO

Total de Registros: 7

Configurar Imp...    Impr. Predefinir    Imprimir    Cerrar

Figura 4.6.20 Reporte de Estatus de Proceso

## Reportes de Sistema

- Reporte de Menús: Este reporte muestra las opciones del menú principal que están disponibles. Además nos muestra el nombre del menú y la ruta donde es lanzado. Ver Figura 4.5.21.
- Reporte de Usuarios: Muestra todos los usuarios validos para el sistema. Nos proporciona información del Departamento al que pertenecen, el estatus existente del usuario, el ID correspondiente y el nombre. Ver Figura 4.5.22.
- Reporte de Permisos: Este reporte permite conocer de forma oportuna todos los permisos que tiene un usuario en particular, Mediante un menú Dropdown se puede seleccionar el usuario en cuestión y conocer, el departamento al que pertenece y el menú al que tiene acceso. Ver Figura 4.5.23.

Opcion	Ruta del Menu	Nombre del Menú
1	GENERALES/PERMISOS	m_permisos
2	REGISTRO/CATALOGOS/CONTACTOS	m_contactos
3	REGISTRO/CATALOGOS/CONCEPTOPAGO	m_conceptopago
4	REGISTRO/CATALOGOS/DEPARTAMENTOS	m_departamento
5	REGISTRO/CATALOGOS/ESTATUS	m_estatus
6	REGISTRO/CATALOGOS/CETES	m_cetes
7	REGISTRO/CATALOGOS/SUARIOS	m_usuarios

Total de Registros: 36

Configurar Imp... Imp. Preliminar Imprimir Cerrar

Figura 4.5.21 Reporte de Menús

Reporte de Usuarios del Sistema

Fecha 25/03/02 Empresa Hipotecaria, S.A. de C.V. Pág 1 de 1

**Reporte de Usuarios del Sistema**

Nombre	Clave Usuario	Departamento	Status
ADMINISTRATOR	ADMINIST	DIRECCION DE TECNOLOGIA DE INFORMACION	ACTIVO
Alejandro	ALEX	SECRETARIO DE CONSEJO	ACTIVO
Proyecto Tesis	AZKABAN	SUBDIRECCION DE COMPUTO	ACTIVO
Azucena	AZUC	DELEGACION TLUANA	ACTIVO
Fernando	FERN	DIRECCION GENERAL	ACTIVO
Josafina Cruz alonso	Josafina	SUBDIRECCION DE COMPUTO	ACTIVO
NIEVES GARCIA	NIEVES	DIRECCION DE TECNOLOGIA DE INFORMACION	ACTIVO
TESIS	TESIS	DIRECCION DE TECNOLOGIA DE INFORMACION	ACTIVO

Configurar Imp... Impr. Preliminar Imprimir Cerrar

Figura 4.5.22 Reporte de Usuarios

Reporte de Permisos por Usuario

Fecha 25/03/02 Empresa Hipotecaria, S.A. de C.V. Pág 1 de 5

**Reporte de Accesos Autorizados**

FERN Fernando  
DIRECCION GENERAL

Clave de Menu	Ruta
2	REGISTROCATALOGOS/CONTACTOS
3	REGISTROCATALOGOS/CONCEPTOPAGO
4	REGISTROCATALOGOS/DEPARTAMENTOS
6	REGISTROCATALOGOS/CETES
8	REGISTROCATALOGOS/MOTIVOSDERECH

Total de Registros: 21

Configurar Imp... Impr. Preliminar Imprimir Cerrar

Figura 4.5.23 Reporte de Permisos

## 4.6 Factibilidad técnica y operativa

En este tema se describirá la forma en que se realizó la puesta en producción, la instalación del software con el cual se desarrolló el sistema y se especificarán algunos puntos importantes para la eficiente operación del sistema.

### 4.6.1 Requerimientos mínimos de hardware y software

La función que tendrá el servidor es la de compartir la información que se encuentra en la base de datos, además que sea eficiente para esta tarea específica.

- Servidor Pentium III (800 Mhz)
- Memoria RAM de 256 MB
- Unidad de CD-ROM 52X interna
- Unidad de respaldo 24 GB (SCSI)
- Disco duro de 20 GB (SCSI)
- 3 bahías disponibles para crecimiento futuro
- 3 ranuras de expansión libres PCI
- Tarjeta de red PCI (Operación Dual 10 BASE T/100 BASE TX)
- 2 puertos seriales y 1 puerto paralelo
- Mouse
- Monitor SVGA color 15"
- UPS con interfaces al servidor (1200 Watts)

El servidor puede ser alimentado por un UPS y contar con una interfaz para protegerlo de forma automática en caso de falla en el suministro eléctrico. Con el fin de poder apagar de manera segura el servidor, evitando la pérdida de información y los posibles daños al equipo.

El servidor debe estar colocado en un área exclusiva y libre de paso, de preferencia el lugar cerrado y con aire acondicionado, para que pueda ser operado sin dificultad cuando sea necesario.

Los requerimientos mínimos para el software son:

- Windows NT Server v4.x
- Windows NT Service Pack 3

También las estaciones de trabajo deben de contar con algunos requerimientos mínimos:

#### **4.6.2 Requerimientos mínimos de hardware y software para las estaciones de trabajo**

Requerimientos de hardware:

- Procesador Pentium II (350 MHz)
- Memoria RAM de 64 MB
- Disco duro de 4 GB
- 2 Ranuras de expansión libres ISA
- 2 Ranuras de expansión libres PCI
- Tarjeta de red PCI (Operación Dual 10BASET/100BASETX)
- Monitor color SVGA
- Mouse (bus)
- 2 Puertos seriales y 1 puerto paralelo

Los requerimientos de software para las estaciones de trabajo, son los siguientes:

- Windows NT Workstation 4.x o
- Windows 95

Es necesario nombrar un administrador del servidor que cumpla con ciertas tareas y responsabilidades para la administración y mantenimiento del servidor.

El administrador deberá asumir las siguientes responsabilidades:

- Administración de usuarios y grupos (altas, bajas y cambios)
- Administrar la seguridad del servidor (asignación de privilegios y contraseñas).
- Administrar los servicios de archivos e impresión
- Respaldos de la información.
- Soporte técnico de primer nivel a los usuarios.
- Instalación y soporte del software de la empresa a los usuarios.
- Asegurar el cumplimiento de las normatividades de sistemas.

Para el buen uso del administrador se requiere a una persona con los conocimientos suficientes de informática para realizar las labores de administración tanto del servidor como del sistema.

El administrador deberá asumir las siguientes responsabilidades para el sistema:

- Administración de estaciones de trabajo, usuarios y grupos (altas, bajas y cambios).
  - Administrar la seguridad del sistema, mediante la asignación de privilegios y contraseñas.
  - Realizar los respaldos de la información, (se recomienda cada semana).
  - Aclaración y atención de dudas, comentarios, funcionamiento y mejoras propuestas por los usuarios del sistema.
  - Brindar soporte técnico a los usuarios del sistema.
  - Instalación y soporte del software del sistema a usuarios.
-

- Llevar un control de los cambios realizados al sistema, así como altas, bajas y cambios en la información de las bases de datos del sistema.

Aun cuando el sistema esta creado bajo los requerimientos del usuario, no están descartadas posibles modificaciones al sistema, con las que puedan aumentar su capacidad o agregar algún requerimiento mas. Es por esta razón que deberá llevarse un control de cambios realizados al sistema, indicando datos importantes como son: la fecha, motivo de la modificación, tipo de modificación, solución propuesta, persona que la autorizó y persona que realizó el cambio.

Este control servirá para poder llevar un histórico del sistema y de cada una de sus modificaciones por fallas, limitaciones del diseño y/o cualquier aumento de necesidades.

Para el buen funcionamiento y crecimiento de un sistema se debe dar un buen mantenimiento, algunos tipos de mantenimiento son los siguientes:

- **Mantenimiento Perfectivo.** Comprende los cambios solicitados al programador del sistema. A medida que se usa el software, se reciben de los usuarios recomendaciones sobre nuevas posibilidades acerca de modificaciones a funciones ya existentes.
- **Mantenimiento Preventivo.** En este tipo de mantenimiento se previenen errores. Este mantenimiento se da cuando se realizan cambios en el software con el fin de mejorar algún proceso. Además del mantenimiento a la información que se maneja para garantizar que los resultados dados por el sistema sean correctos.
- **Mantenimiento Adaptativo** Se debe a cambios en el ambiente del programa y a la adaptación de nuevas unidades o módulos. De este mantenimiento pueden derivarse los siguientes mantenimientos:



- **Mantenimiento Aumentativo.** Este tipo de mantenimiento se da cuando se incluyen nuevas funciones que no se contemplan al inicio del desarrollo del sistema y surgen como una necesidad del usuario.
- **Mantenimiento Tecnológico.** Se da debido a los cambios importantes en la informática (hardware y software), es decir al avance en los sistemas operativos, lenguajes de programación y nuevas generaciones de hardware, con lo cual se mejoran las herramientas de trabajo, las versiones de software, equipos periféricos y otros elementos de sistemas.
- **Mantenimiento Correctivo.** El proceso que incluye el diagnóstico y corrección de uno o más errores en el sistema se denomina mantenimiento correctivo. Durante el uso del sistema se encuentran errores, los cuales deben ser informados al equipo de desarrollo.

En la figura 4.6.2.1 se muestran más claramente los diferentes tipos de mantenimiento por los que pasa un sistema:

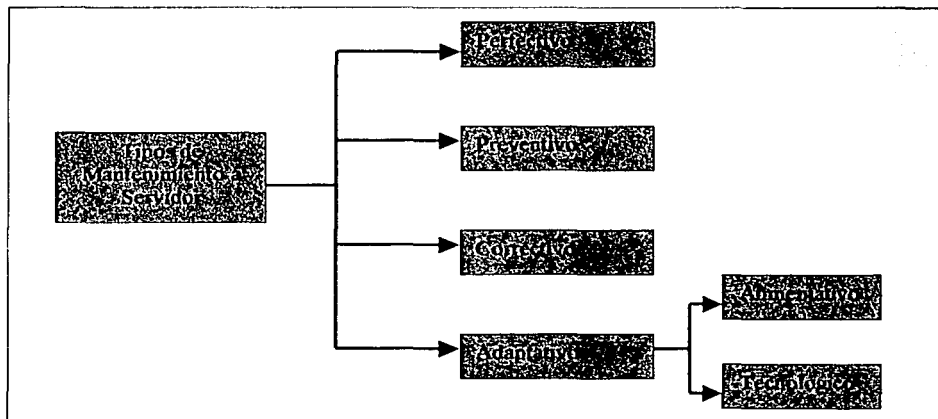


Figura 4.6.2.1 Tipos de mantenimiento al servidor

Debido a las características del programa y que tendrá un tiempo de adaptabilidad para su aceptación total por parte del usuario, el mantenimiento que se utilizará será de tipo perfecto al principio, es decir en base a las recomendaciones de los usuarios. Tratando de satisfacer estos requerimientos se realizarán los cambios necesarios.

Cuando el sistema haya alcanzado el funcionamiento deseado, se realizarán mantenimientos preventivos, con la finalidad de garantizar la información, los resultados óptimos del sistema y la prevención de errores.

El sistema tendrá que adecuarse a las nuevas modificaciones y/o cambios, para lo cual deberán realizarse mantenimientos adaptativos.

En cuanto a los mantenimientos correctivos, estos serán realizados en cuanto se presente un problema o error en el sistema, para lo cual se pide a los usuarios del sistema informen cualquier anomalía.

#### **4.6.3 Análisis del Costo y Venta del Sistema**

##### **Estimación de costos**

La estimación de costos depende en gran medida del personal experimentado para proporcionar presupuestos adecuados y óptimos, además, para que puedan visualizar con anticipación cuáles serán los eventos que se presentarán a lo largo del desarrollo.

En caso contrario se puede tener un error muy grave con consecuencias considerables, como las pérdidas en el equipo de desarrollo o la pérdida de las licitaciones, por lo que es necesario considerar los elementos variantes como el personal, el tiempo de desarrollo, técnicas utilizadas, entre otros.

Para esto, existen procedimientos sistematizados que reducen el grado de riesgos, para esto, es recomendable realizar un estudio preliminar y realizar una revisión de requisitos del sistema. Dentro de los factores que nos ayudan a considerar los costos están los siguientes:

- Tamaño del programa.
- Complejidad del producto.
- Tiempo disponible: se refiere al esfuerzo invertido.
- Confiabilidad requerida: dependiendo de este factor se toman las decisiones sobre las estrategias a emplear a lo largo del desarrollo.
- Nivel tecnológico: se refleja en los elementos utilizados.

En la evaluación de costos, la técnica más utilizada es la del juicio experto, de tipo jerárquico descendente. Se basa en el conocimiento anterior y en el sentido comercial. Esta técnica nos impide tomar decisiones de manera confiada y que se subestimen factores que no se han considerado.

Para obtener el costo total del sistema se analizaron los siguientes tres puntos:

- Costos por línea de código.
- Costo total del proyecto.
- Costo por Hora/Hombre, teniendo los siguientes tipos:
  - Líder de Proyecto
  - Programador Senior
  - Especialista Administrativo
  - Programador Junior

Para el costo por línea de código, tenemos que el costo se encuentra entre 20 y 30 dólares. Si consideramos que nuestro sistema consta de aproximadamente 2,000 líneas de código.

Haciendo las operaciones correspondientes y tomando a 20 dólares por línea, obtenemos un costo de 40,000 dólares al tipo de cambio actual.

Para el tercer punto, se tiene que el costo Hora/Hombre se calcula con base al tiempo que invierte cada persona que participa en el proyecto, por lo tanto se puede establecer un costo por persona como se muestra en la figura 4.6.3.1:

Tipo	Costo por (Día de 8 horas) en dólares
Lider de Proyecto	\$960
Programador Señor	\$800
Administrador del Proyecto	\$720
Programador Junior	\$400

Figura 4.6.3.1 Tabla de Costo Hora/Hombre.

Los proyectos de software no deben ser abordados por grupos grandes y homogéneos de ingenieros de software, ya que hacen que el tiempo consumido en comunicaciones entre los miembros del grupo sea mayor que el tiempo que se pasa programando. Además, suele ser imposible dividir un sistema en una gran cantidad de unidades independientes. Esto da como resultado que, si se emplea un grupo grande de programación, cada miembro sea responsable de una unidad de programa cuya interfaz con el resto del sistema es compleja. En consecuencia, la probabilidad de errores en las interfaces es alta y las pruebas del software son difíciles y lentas. El tamaño de los grupos de programación debe ser relativamente pequeño entre dos y ocho miembros.

## Grupos con jefe de programadores

El desarrollo de este enfoque fue motivado por varios aspectos:

- Se tiende a asignar personal relativamente inexperto a los proyectos.
- Gran cantidad del trabajo de programación es administrativo, lo que da como resultado el almacenamiento y mantenimiento de un enorme volumen de información.
- La comunicación de múltiples canales consume tiempo y, por tanto, reduce la productividad del programador.

El grupo con jefe de programadores se basa en el empleo de personal con experiencia y talento como jefes de programación; se les brinda apoyo administrativo mediante procedimientos manuales, y canalizando toda la comunicación a través de uno o dos individuos.

El núcleo de un grupo con jefe de programadores consta de los siguientes miembros:

- Un programador jefe, que tiene experiencia y esta altamente calificado. Es el responsable del diseño, programación, prueba e instalación del sistema en desarrollo.
- Un programador auxiliar, capaz y experimentado. Este trabaja con el programador jefe y debe poder ocupar su puesto en caso necesario. Su función principal es proporcionar apoyo por medio del desarrollo de casos de prueba y análisis para comprobar el trabajo del programador jefe.
- Un bibliotecario, cuya función es asumir todas las funciones administrativas relacionadas con un proyecto. El bibliotecario se sirve de un sistema de biblioteca automatizado.

Según el tamaño y tipo de la aplicación, al grupo se pueden sumar otros expertos de manera temporal o permanente. Estos podrían ser:

- Un administrador del proyecto que releva al jefe de programadores en las tareas administrativas.
- Un fabricante de instrumentos, responsable de producir los instrumentos de software para apoyar el proyecto.
- Un editor de documentación, que toma la documentación del proyecto escrita por el programador jefe y el programador auxiliar y la prepara para su publicación.
- Un experto en el lenguaje y en el sistema que este familiarizado con la idiosincrasia del lenguaje de programación y con el sistema que se emplee. Su función es asesorar al programador jefe sobre la utilización de esas características.
- Un probador, cuya tarea es desarrollar casos de prueba objetivos para comprobar el trabajo del programador jefe.
- Uno o mas programadores auxiliares, encargados de la codificación de un diseño especificado por el programador jefe. Estos programadores son necesarios cuando el tamaño del proyecto es tal que el trabajo de programación detallado no puede ser realizado solo por el jefe de programadores y por el programador auxiliar.

El principal objetivo de utilizar un grupo con jefe de programadores es mejorar la productividad; estudios realizados revelan que un grupo con jefe de programadores es aproximadamente el doble de productivo que los grupos organizados de otra manera.

La estimación de los precios, tomando en cuenta un total de 100 días (tiempo estimado para la liberación del sistema) los costos, que al igual que los tiempos, fueron tomados de algunas encuestas hechas a personas expertas en el área de planeación de sistemas, se especifican en la figura 4.6.3.2:

ETAPA	RECURSOS	DÍAS	COSTO/DÍA	TOTAL
Análisis	3	30	\$400	\$36,000
Diseño	3	20	\$400	\$24,000
Codificación	2	20	\$350	\$14,000
Pruebas	2	10	\$350	\$14,000
Liberación	2	10	\$350	\$14,000
Documentación	2	5	\$350	\$3,500
Capacitación	2	5	\$350	\$3,500
<b>Total</b>		<b>100</b>	<b>\$2550.00</b>	<b>\$109,000.00</b>

Figura 4.6.3.2 Estimación de costos del sistema de automatización de la cobranza.

En base a las estimaciones del punto uno y tres, además de hacer un ajuste a los costos anteriores y basándonos en el tiempo de desarrollo del sistema, tenemos que para el punto dos el costo del proyecto sería de \$121,500 dólares.

La empresa a la cual se le esta desarrollando el sistema, ya cuenta con la infraestructura de red necesaria para la implementación del sistema, así como el hardware y software necesarios, por lo tanto no se incluyen estos conceptos en el análisis de costos.

El cliente para el cual se desarrolla el sistema tiene como política adquirir totalmente los derechos de los programas realizados, es decir, requiere que le sea entregado la totalidad del código del sistema, manuales de usuario y manuales técnicos.

Por lo tanto la operación de compra-venta del sistema se realizará en forma total.

#### 4.6.4 Implantación del Proyecto

##### Planeación del proyecto

La administración efectiva de un proyecto de software depende de la planeación detallada de su avance, anticipando problemas que puedan surgir y preparando con anticipación soluciones tentativas a ellos. Se supondrá que el administrador del

proyecto es responsable de la planeación, desde la definición de requisitos hasta la entrega del sistema terminado. No se analizará la planeación que implica la estimación de la necesidad de un sistema de software, la viabilidad de producir tal sistema y la asignación de prioridad al proceso de su producción.

La planeación que se estudia aquí es la requerida para un sistema grande de programación que necesita múltiples grupos para abordar el proyecto y poderlo terminar en un tiempo dado. Sin embargo, gran parte del estudio es válido para la producción de sistemas más pequeños a cargo de un solo grupo de programación, aunque, para tales sistemas, la planeación puede ser menos formal.

Metzger (1973) define que un plan de proyecto se compone de 11 secciones diferentes. Estas son:

1. Panorama. Hace una descripción general del proyecto, detalla la organización del plan y resume el resto del documento.
2. Plan de fases. Se analiza el ciclo de desarrollo del proyecto: análisis de requisitos, fase de diseño de alto nivel, fase de diseño de bajo nivel, etc. Asociada con cada fase debe haber una fecha que especifique cuando se debe terminar esa fase y una indicación de como se pueden solapar las distintas fases del proyecto.
3. Plan de organización. Se definen las responsabilidades específicas de los grupos que intervienen en el proyecto.
4. Plan de pruebas. Se hace un esbozo general de las pruebas requeridas y de las herramientas, procedimientos y responsabilidades para realizar las pruebas del sistema. No se incluyen casos de prueba específicos.
5. Plan de control de modificaciones. Se establece un mecanismo para aplicar las modificaciones que se requieran a medida que se desarrolle el sistema.
6. Plan de documentación. Su función es definir y controlar la documentación asociada con un proyecto.



7. Plan de capacitación. Se describe la preparación de los programadores que participan en el proyecto y las instrucciones a los usuarios para la utilización del sistema que se les entrega.
8. Plan de revisiones e informes. Se analiza como se informa del estado del proyecto. Aquí se definen las revisiones formales asociadas con el avance del proyecto.
9. Plan de instalación y operación. Se describe el procedimiento para instalar el sistema en la localidad del usuario.
10. Plan de recursos y entregas. Se resumen los detalles críticos del proyecto: fechas programadas, marcas de logros y todos los artículos que se deben entregar bajo contrato.
11. Índice. Se muestra en donde encontrar las cosas dentro del plan.

Otra técnica es la de GANTT, que nos ayuda a hacer una buena planificación de las tareas a realizar en el sistema, el tiempo estimado para cada una de ellas, considerar un tiempo de holgura y determinar la secuencia (si la hay) en que deben llevarse a cabo las actividades. Nos ayuda, además, a establecer metas y objetivos tomando en cuenta los recursos y la habilidad con la que se cuenta para realizar cada actividad.

Para nuestro caso las tareas definidas se mencionan a continuación, así como el tiempo estimado en términos de porcentaje de cada tarea, cabe señalar que estos tiempos fueron tomados de algunas encuestas hechas a personas expertas en el área de planeación de sistemas:

TAREA	TIEMPO ESTIMADO (EN PORCENTAJE)
ANÁLISIS	30%
DISÑO	20%
CODIFICACIÓN (PROGRAMACIÓN)	20%
PRUEBAS (INCLUYENDO LA PREPARACIÓN DE DATOS DE PRUEBA)	10%
LIBERACIÓN (IMPLEMENTACIÓN E INSTALACIÓN DEL PRODUCTO)	10%
DOCUMENTACIÓN	5%
CAPACITACIÓN A USUARIOS	5%

Figura 4.6.4.1 Tareas definidas para el sistema de automatización del proceso de cobranza de una empresa hipotecaria, con su tiempo estimado.

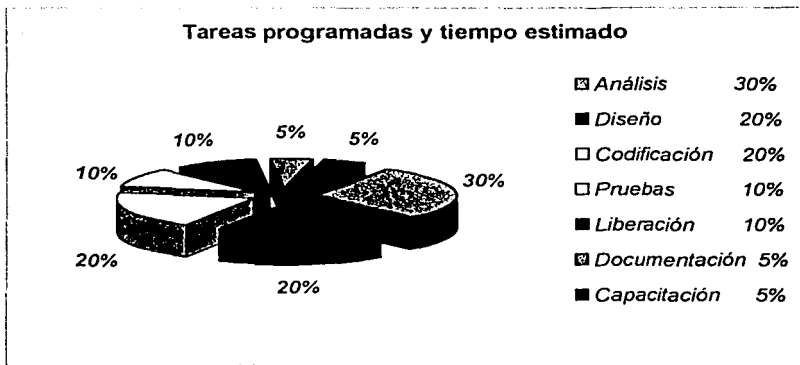


Figura 4.6.4.2 Diagrama de pastel para clarificar el tiempo estimado

El producto final contará con cuatro meses de garantía, se ceden derechos sobre el código y la documentación. A los precios se agregan los de tipo legal: bajo buen cobro en caso de cheques y las penalizaciones en caso de retraso en el tiempo.

Los precios de hardware y software son sólo actualizaciones, ya que la empresa contaba ya con algunas licencias, a las cuales se le agregaron sólo algunos componentes y herramientas necesarias para el buen funcionamiento del sistema.

Para finalizar este capítulo, presentaremos el diagrama de GANTT que esquematiza mejor la planeación realizada para el alcance de objetivos.

Id	Nombre de tarea	Duración	Comienzo	Fin	Predecesoras	Nombres de los recursos
0	Proyedt	100 días	lun 01/04/02	vie 16/08/02		
1	Análisis	30 días	lun 01/04/02	vie 10/05/02		
2	Entrevistas	7 días	lun 01/04/02	mar 09/04/02		
3	Definición del proyecto	9 días	mié 10/04/02	lun 22/04/02		
4	Estudio de factibilidad	7 días	mar 23/04/02	mié 01/05/02	3	
5	Propuesta de solución	7 días	jue 02/05/02	vie 10/05/02	4	
6	Diseño	20 días	lun 13/05/02	vie 07/06/02	2	
7	Diagramas de contexto	4 días	lun 13/05/02	jue 18/05/02		
8	Diagramas de datos	4 días	vie 17/05/02	mié 22/05/02		
9	Diagramas de entidad/relación	4 días	jue 23/05/02	mar 28/05/02		
10	Decomano de datos	4 días	mié 29/05/02	lun 03/06/02		
11	Normalización	4 días	mar 04/06/02	vie 07/06/02		
12	Codificación	20 días	lun 10/06/02	vie 05/07/02	6	
13	Algoritmos	10 días	lun 10/06/02	vie 21/06/02		
14	Programación	10 días	lun 24/06/02	vie 05/07/02		
15	Pruebas	12 días	lun 08/07/02	mar 23/07/02	12	
16	Elaborar datos de prueba	2 días	lun 08/07/02	mar 09/07/02		
17	Unitarias y de Integración	2 días	mié 10/07/02	jue 11/07/02	16	
18	Regresión y Volumen	2 días	vie 12/07/02	lun 15/07/02	17	
19	Caja negra y blanca	2 días	mar 16/07/02	mié 17/07/02	18	
20	Estáticas y Funcionales	2 días	jue 18/07/02	vie 19/07/02	19	
21	Estructurales y de Usuario	2 días	lun 22/07/02	mar 23/07/02	20	
22	Liberación	10 días	mié 24/07/02	mar 06/08/02	15	
23	Instalación en producción	5 días	mié 24/07/02	mar 30/07/02		
24	Carga de información	5 días	mié 31/07/02	mar 06/08/02	23	
25	Documentación	4 días	mié 07/08/02	lun 12/08/02	22	
26	Manual Técnico	2 días	mié 07/08/02	jue 08/08/02		
27	Manual de usuario	2 días	vie 09/08/02	lun 12/08/02	26	
28	Capacitación	4 días	mar 13/08/02	vie 16/08/02	25	
29	Capacitación a usuarios	4 días	mar 13/08/02	vie 16/08/02		

Figura 4.6.4.3 Diagrama de Gantt

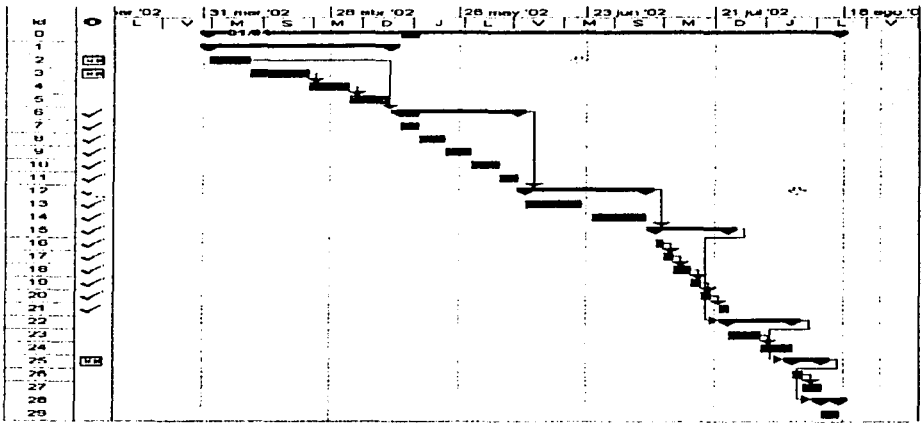


Figura 4.6.4.4 Diagrama de Gantt



## CONCLUSIONES

Desde el surgimiento de las computadoras, el trabajo de almacenamiento de información ha disminuido el tiempo hombre invertido en búsquedas y ha mejorado su disponibilidad. El hecho de sistematizar un proceso manual, mejora la calidad del trabajo y permite al empleado realizar otras actividades además de las que desempeña comúnmente.

Este sistema servirá para ayudar a agilizar un proceso que traerá un gran beneficio en la empresa debido a que se realizarán los trámites pertinentes con una mayor rapidez y se podrá obtener información actualizada en el momento.

Es un poco desalentador, aunque comprensible, el hecho de que la gente de edad avanzada se resista a la implantación de sistemas, por temor al cambio o a ser sustituidos, argumentando que ellos ya no lo necesitan y que eso se lo dejan a la gente joven; pero esta actitud provoca un retraso en el avance, no sólo de las empresas sino del país en general, ya que frena al desarrollo del conocimiento e incrementa la ignorancia.

Es necesario informarse e incluso asesorarse, veraz y ampliamente, sobre todo lo que implica la adquisición de un bien inmueble a través de un crédito hipotecario, antes de realizar cualquier trámite de apartado, ya que el involucrarse en adeudos difíciles de liquidar limitará la calidad de vida del adquirente.

En la actualidad, el área de sistemas de cualquier empresa se ha convertido en una entidad vertebral para su desarrollo, pues lo mismo lo necesita un despacho jurídico que un laboratorio de investigación, y producto de las necesidades de dichas instituciones es que los programas de cómputo han tenido un despunte de actualización y aplicaciones tan diversas, que resulta difícil concebir una sola persona

con el conocimiento de todas y cada una de las herramientas disponibles en el mercado.

Con el presente trabajo de tesis, hemos retomado algunos temas de la carrera que por algún momento quedaron rezagados, por enfocarnos a una plataforma específica por requerirlo así nuestro desempeño profesional. También nos ha permitido conocer nuevos temas que durante la carrera no fueron impartidos en ninguna materia.

La convivencia con el equipo de trabajo ha sido grata, más aún porque cada uno nos desempeñamos en áreas distintas de trabajo, pero ello nos permite conocer diversos puntos de vista.

Hemos desarrollado nuevas formas de trabajo, y nuevos conceptos técnicos que nos han facilitado la participación en ella. El aprendizaje fue muy sencillo, y la forma de compartir conocimientos fue ágil y divertida. La buena comunicación en el equipo facilitó el trabajo.

La pluralidad de carreras y de personas que la Universidad tiene, enriquece cualquier trabajo que en equipo puede realizarse, y prueba de ello es el trabajo de esta Tesis, en la cual hemos colaborado personas de diferentes generaciones, y hemos compartido no sólo conocimiento, sino buenas experiencias.

Agradecemos al Programa de Apoyo a la Titulación la oportunidad que nos brinda para concluir con nuestra carrera, ya que al ingresar al área laboral, no contamos con el tiempo necesario para llevar a cabo este proceso, sin embargo, este programa ofrece horarios factibles y asesorías de buena calidad, lo cual hace sencillos tanto los trámites como la elaboración de la tesis.

El programa de apoyo a la Titulación nos da la oportunidad de lograr concluir nuestros estudios profesionales a personas que actualmente nos encontramos ejerciendo la

---

profesión. El seguimiento y corrección del trabajo, nos permite desarrollar un trabajo con calidad, además de guiarnos en lo que espera de nosotros la Universidad.

Es un buen método para la titulación de los egresados que por una u otra razón han dejado del lado su proceso de titulación, este programa ayuda a los pasantes a comprometerse a terminar su proceso de titulación y lo hace de una manera sencilla pero profesional.

El Programa de apoyo a la titulación otorga la facilidad de realizar trámites, de asistir a las asesorías sin dejar de realizar nuestras actividades laborales y de seguimos preparando.

En cuanto a la carrera podemos concluir que se tienen buenas bases para la formación de ingenieros, aunque algunos de los cursos no cumplen por completo con las expectativas o algunas veces son incompletos, sugerimos que se tenga en consideración esto para una formación más completa que la que se tiene actualmente.

Consideramos que los nuevos planes de estudios que la UNAM sugiere para las generaciones actuales, deben de ir a la vanguardia de las nuevas propuestas y que tanto los egresados, como los actuales estudiantes, no sólo debemos de quedarnos en lo que la escuela nos da, sino emprender una carrera autónoma.

La Universidad nos ha dado las bases necesarias para poder entrar a un mercado laboral cada vez más competitivo, y depende de cada individuo el poner en alto el nombre de la UNAM haciendo lo mejor en su trabajo cotidiano y aplicando los conocimientos adquiridos durante dicha formación.

La formación que hemos recibido en La Universidad, nos ha hecho capaces de enfrentar los nuevos retos y de llegar a soluciones óptimas, muestra de ello es la elaboración de esta tesis, mediante la cual hemos aprendido nuevas aplicaciones, pero que no nos ha dejado al margen de los nuevos desafíos.



Un ingeniero, no sólo debe ser una persona con habilidad técnica, lógica o analítica, sino también debe tener una formación humana, lo que lo hace sensible a las necesidades del ambiente en el que se desarrolla.

Durante el desarrollo de esta tesis, nos hemos dado cuenta de la necesidad que tenemos de formarnos, día con día, para poder ofrecer a la sociedad un servicio completo y responsable. El software crece rápidamente al igual que los medios de comunicación, lo cual es un empuje a continuar con nuestra carrera y no pensar que los estudios ya concluyeron.

## MANUAL DE USUARIO

El objetivo del presente manual es proporcionar al usuario la información básica y necesaria para conocer la estructura y funcionalidad del sistema. En éste se explicarán y darán a conocer las opciones que tiene el sistema llamado Cobranza, de una forma gráfica y textual.

### Como acceder al Sistema de Cobranza

Para acceder al sistema es necesario seleccionar el icono de acceso directo que se encuentra en el escritorio (figura A.1). Al ejecutar la aplicación nos mostrará una barra de estado (figura A.2) en la cual el usuario verá el progreso de la carga del mismo (conexión a la base de datos, validación de la clave de usuario y contraseña a las opciones del menú).



Figura A.1 Icono de Acceso Directo

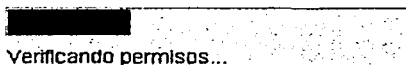


Figura A.2 Carga del Programa

Posteriormente el usuario tendrá que introducir su login y su password, como se muestra en la figura A.3. Cabe señalar que esto se realiza como medida de seguridad para restringir el acceso a personal no autorizado.

En caso de que el login o el password no coincidan con los usuarios autorizados para el uso del sistema, se enviará un mensaje de error (figura A.4) y el usuario deberá ponerse en contacto con el administrador del Sistema de Cobranza.

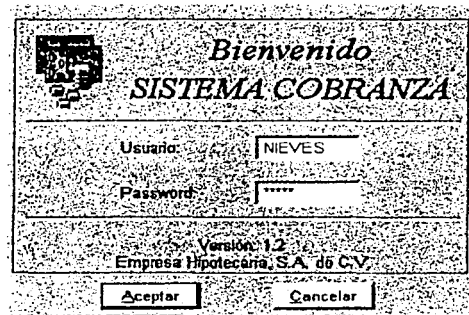


Figura A.3 Ventana Inicial

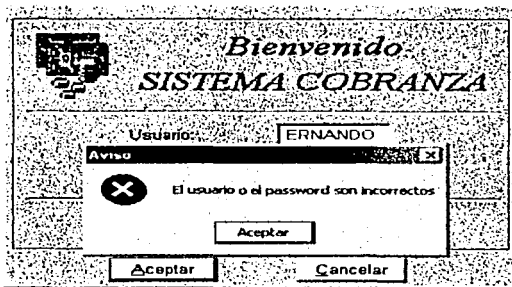


Figura A.4 Ventana de Error por Usuario o Password incorrectos

Una vez que el usuario ha accedido exitosamente al sistema se mostrará la ventana principal, en donde se observa la barra de herramientas, como se ve en la figura A.5.

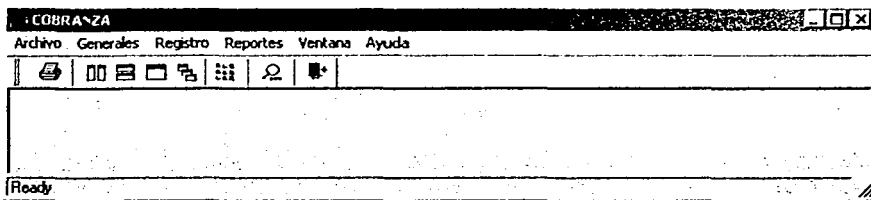


Figura A.5 Ventana Principal

De acuerdo a los privilegios que tenga el usuario sobre los menús este podrá acceder parcial o totalmente a las opciones de los mismos.

A continuación se mencionarán las opciones comprendidas en los cada uno de los menús:

- Archivo.- Impresión Preliminar, Configurar Impresora y Salir.
- Generales.- Línea Empresarial.
- Registro.- Catálogos (incluye: Contactos, Conceptos de Pago, Departamentos, Estatus, Cetes, Usuarios, Motivos de Rechazo, Menús y Conceptos), Accesos Usuarios, Captura de Pagos y Asocia Pagos.
- Reportes.- Pagos Asociados (por usuario y por fecha), Pagos Pendientes, Intermediarios Financieros, Conceptos de Pago, Departamentos, Estatus de Proceso, Tasa Cetes Vigentes, Tipos de Rechazo, Tipos de Pago y Sistema (Menús, Usuarios y Permisos).
- Ventanas.- Vertical, Horizontal, Mosaico, Cascada, Organizar Iconos y Barra de Herramientas.
- Ayuda.- Como Usar la Ayuda, Acerca del sistema y Actualiza Permisos.

## Funciones Generales.

Antes de continuar, conviene hacer un breve paréntesis para describir las funciones generales que ayudarán al usuario a un mejor manejo/conocimiento del sistema:

- Para desplazarse de un campo a otro se puede hacer mediante la tecla TAB o con el mouse al elegir el campo, veamos la figura A.6

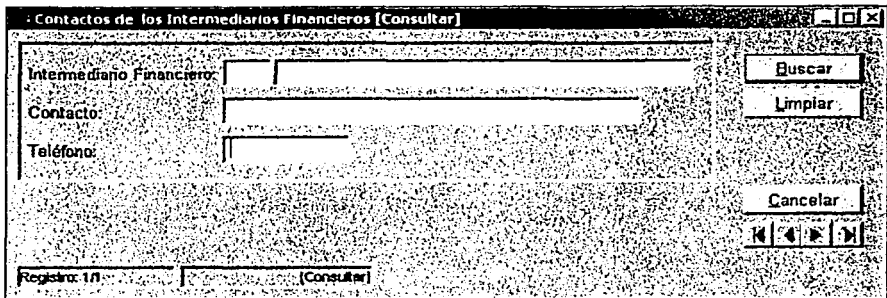


Figura A.6 Desplazamiento con el tabulador

- Cuando se intente dar de alta un nuevo registro, se deberán llenar todos los campos requeridos, de lo contrario se enviará un aviso que indicará que falta información para el nuevo registro.

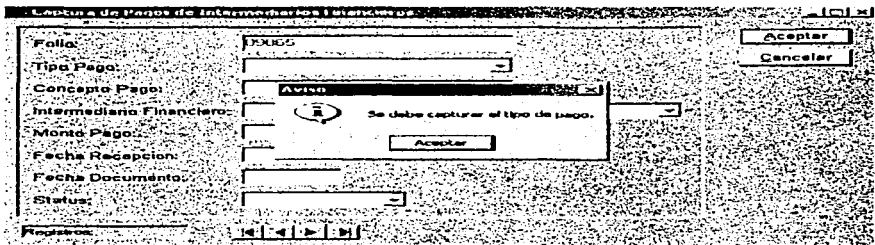


Figura A.7 Aviso de omisión de captura

- Al utilizar un campo combo, que es aquel que despliega una lista de valores permitidos en ese campo en especial, existen varias formas para abrirlo, por ejemplo:
  - Haciendo click con el mouse en la flecha que se encuentra en el extremo derecho del combo.
  - Una vez posicionado en el campo del combo se oprime la tecla F4 o las teclas Alt + ↑ ó ↓.

Una vez abierto el combo se podrá elegir el valor deseado utilizando ↑ ó ↓, o con el puntero del mouse. Así mismo, no es necesario abrir el combo ya que uno puede desplazarse por medio de ↑ ó ↓ o si se conoce la primera letra de la opción.



Figura A.8 Campo con opción Combo

Comprende las opciones Impresión Preliminar, Configurar Impresora y Salir.

## Generales

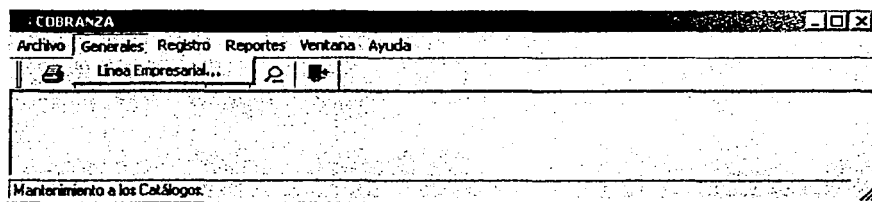


Figura A.9 Menú Generales.

Este menú contiene la opción "Linea Empresarial" la cual permite importar desde un archivo (con formato tipo texto, separado por tabuladores y sin encabezados), todos los movimientos registrados en la cuenta bancaria de la Empresa Hipotecaria para posteriormente asociarlos a los pagos efectuados con los documentos capturados. Dicho archivo es generado a partir de una conexión directa con el banco que maneja las cuentas de la Empresa Hipotecaria.

Para la importación es necesario introducir un disco de 3 ½ en la unidad de disco de la PC, donde se encuentra la información, y oprimir el botón "Archivo", veamos la figura A.10. De lo contrario, el sistema enviará un mensaje de error como se muestra en la figura A.11 y nos permitirá importar el archivo desde otra localidad.

Si usted desea salir de esta ventana solo basta oprimir el botón "Cerrar".

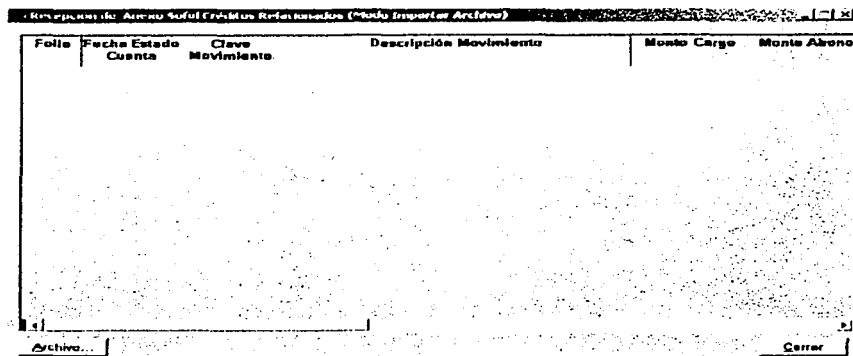


Figura A.10 Ventana de la opción Línea Empresarial

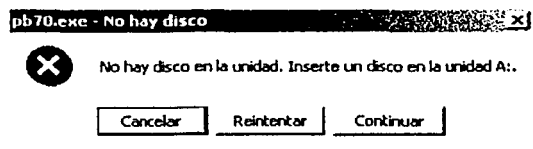


Figura A.11 Mensaje de Error al omitir el disco en la unidad A

Una vez que se importaron los datos exitosamente, usted podrá visualizar la siguiente información:

- Folio
- Fecha del Estado de Cuenta
- Clave del Movimiento
- Descripción del Movimiento
- Monto Cargo
- Monto Abono
- Saldo Cta.
- Claves Estado Movimiento
- Folio Transferencia
- Operador
- Referencia
- Sucursal
- Banco Origen
- Clave Devolución
- Hora
- Status



## Registro.

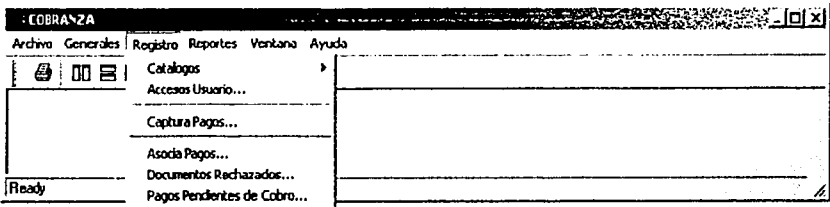


Figura A.12 Submenú Registro

En este menú se divide en tres secciones:

- La primera es donde se actualizan los catálogos y modifican los accesos de los usuarios a las opciones de los menús.

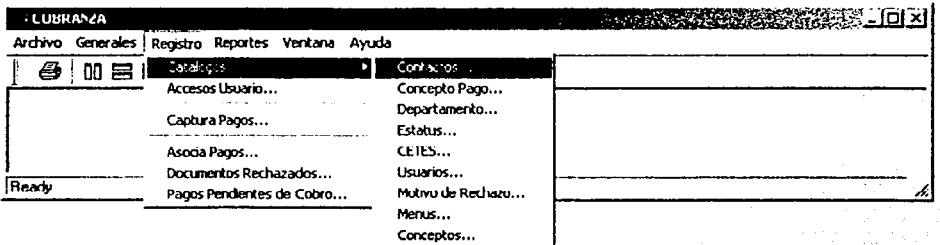


Figura A.13 Opciones del submenú Catálogos

Como se muestra en la figura A.13 el submenú de Catálogos comprende las opciones de :

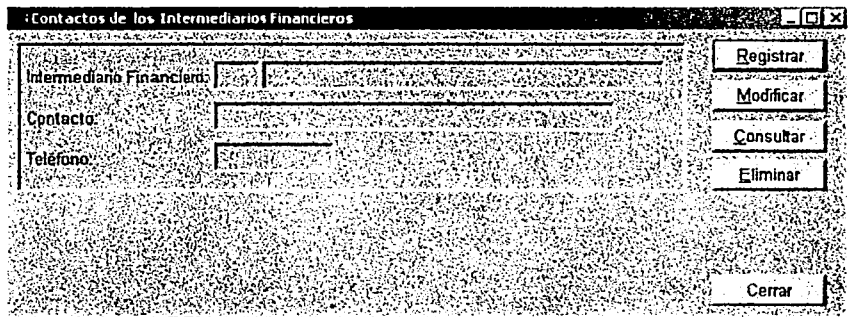
- Contactos
- Concepto Pago
- Departamento
- Estatus
- CETES
- Usuarios
- Motivo Rechazo
- Menús
- Conceptos

Las operaciones que es posible efectuar sobre cualquiera de ellos son cuatro:

- Registrar
- Modificar
- Consultar
- Eliminar

### Cómo dar de alta registros en un catálogo

En general el ingreso de nuevos registros en los catálogos existentes se efectúa a través de una ventana como la que se muestra en la figura A.14.



The screenshot shows a window titled "¡Contactos de los Intermediarios Financieros". Inside the window, there is a form with three input fields. The first field is labeled "Intermediario Financiero", the second is labeled "Contacto", and the third is labeled "Teléfono". To the right of the form, there are five buttons: "Registrar", "Modificar", "Consultar", "Eliminar", and "Cerrar".

Figura A.14 Registro de un nuevo contacto de Intermediario Financiero.

Al oprimir la opción **Registrar** podrá visualizar todos los campos necesarios para dar de alta un registro, cabe señalar que el sistema realizará las validaciones necesarias y, en su caso, desplegará un mensaje de error solicitando la correcta captura de la información.

A continuación, se presenta un ejemplo, en este caso se agregará un nuevo departamento, en el catálogo del mismo nombre:

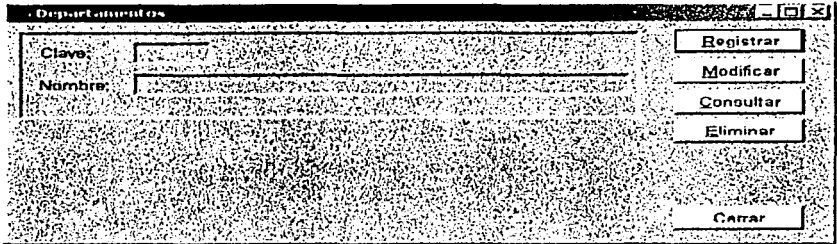


Figura A.15 Ventana principal del Catálogo Departamentos

Una vez seleccionado el botón "Registrar" aparece una ventana como la que se observa en la figura A.16.



Figura A.16 Agregando un nuevo departamento.

El botón "Aceptar", que se encuentra en la parte superior derecha de la ventana, nos permitirá salvar el nuevo registro; una vez que los campos de Clave y Nombre hayan sido capturados. De lo contrario nos enviará un mensaje de error. En la figura A.17 se muestra el mensaje cuando la operación fue exitosa.



Figura A.17. Aviso de información actualizada

En cambio, el botón "Cancelar" finalizará la operación, regresando el control del sistema a la ventana principal del catálogo de Departamentos sin realizar ninguna operación, figura A.15.

#### Cómo modificar un registro en un catálogo.

El usuario puede cambiar la información de un registro de un catálogo, de la siguiente forma: una vez elegido el catálogo solo basta seleccionar la opción "Modificar" de la ventana principal y a continuación aparecerá otra ventana como la que se observa en la figura A.18. En la cual primero es necesario buscar el registro para posteriormente modificarlo.

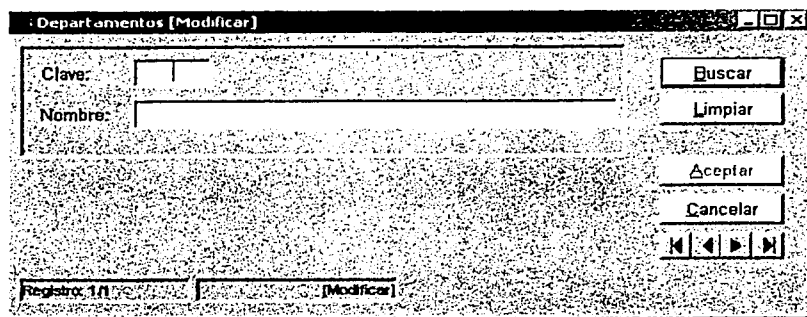


Figura A.18 Opción Modificar del catálogo Departamentos

Siguiendo con el ejemplo anterior, modifiquemos el registro anteriormente capturado:

Primero debemos buscar el registro deseado, para ello es necesario teclear ya sea la clave o el nombre del departamento y en seguida oprimir el botón "Buscar".

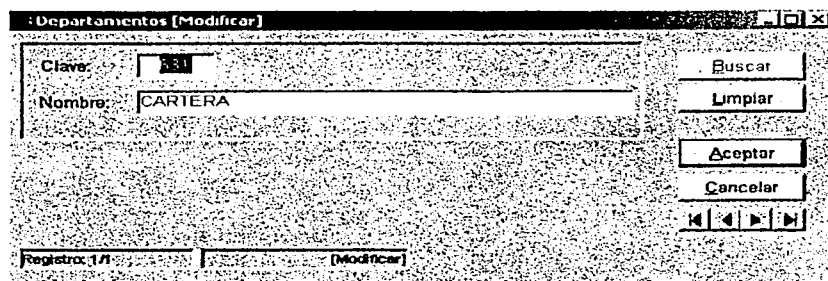


Figura A.19 Búsqueda del registro a modificar

Es posible que usted no recuerde ni la clave o el nombre, no se preocupe, para encontrar el registro solo es necesario oprimir los botones de flecha izquierda o derecha, los cuales le mostrarán registro por registro.

Cuando el sistema le devuelva el resultado de la búsqueda, usted visualizará el registro con todos los campos, y después de realizar los cambios convenientes, se debe presionar el botón "Aceptar" para salvar la información, en caso contrario oprimir cancelar para volver a la ventana principal del catálogo.

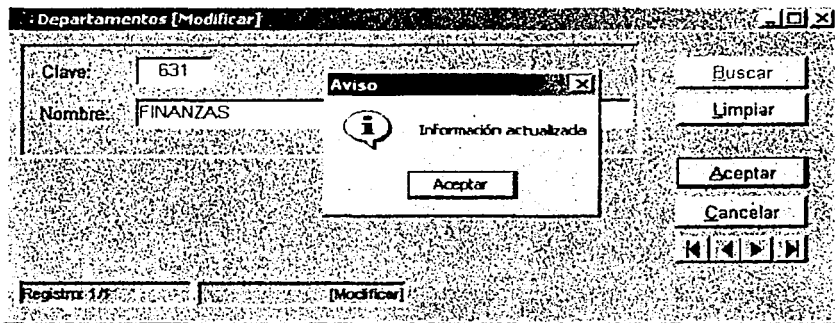


Figura A.20 Aviso de actualización exitosa

### Consulta de registros.

La búsqueda de cualquier registro se realiza de la siguiente manera.

En la pantalla principal del catálogo deseado deberá oprimirse el botón "Consultar", a continuación se presenta una ventana con las opciones: **Buscar**, **Limpiar** y **Cancelar**. Veamos el ejemplo de la figura A.21.

Una vez que usted haya tecleado ya sea la clave o cualquier otro campo indicado en la misma ventana, oprima "Buscar" para realizar la consulta por el/los campo(s) indicado(s) y en seguida el sistema devolverá el resultado de la consulta, figura A.22.

Cuando el sistema le devuelva el resultado de la búsqueda, usted visualizará el registro con todos los campos, y después de realizar los cambios convenientes, se debe presionar el botón "Aceptar" para salvar la información, en caso contrario oprimir cancelar para volver a la ventana principal del catálogo.

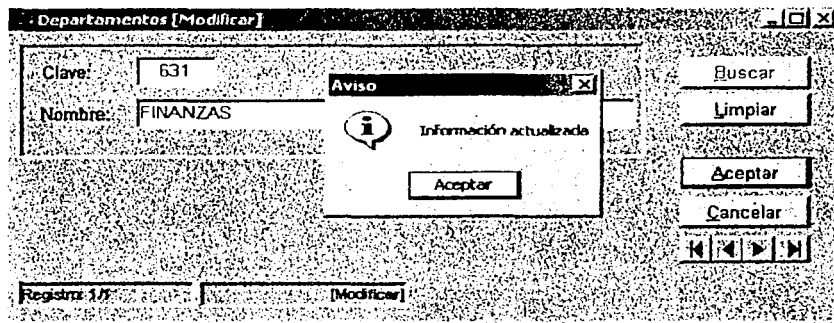


Figura A.20 Aviso de actualización exitosa

### Consulta de registros.

La búsqueda de cualquier registro se realiza de la siguiente manera.

En la pantalla principal del catálogo deseado deberá oprimirse el botón "Consultar", a continuación se presenta una ventana con las opciones: **Buscar**, **Limpiar** y **Cancelar**. Veamos el ejemplo de la figura A.21.

Una vez que usted haya tecleado ya sea la clave o cualquier otro campo indicado en la misma ventana, oprima "Buscar" para realizar la consulta por el/los campo(s) indicado(s) y en seguida el sistema devolverá el resultado de la consulta, figura A.22.

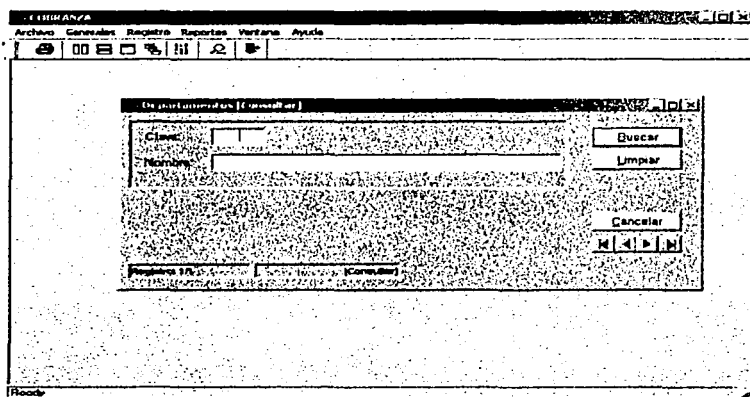


Figura A.21 Ventana de consulta del catálogo Departamentos

Pulse "Limpiar" para realizar otra búsqueda.

Otra forma de realizar una consulta de un catálogo es oprimiendo el botón "Buscar" y avanzar con las flechas izquierda y derecha para revisar registro por registro, figura A.23.

Para salir de consultas oprimir "Cancelar".



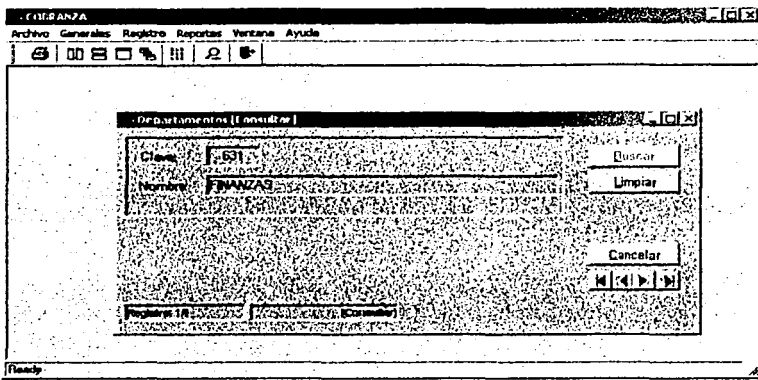


Figura A.22 Resultado de la búsqueda

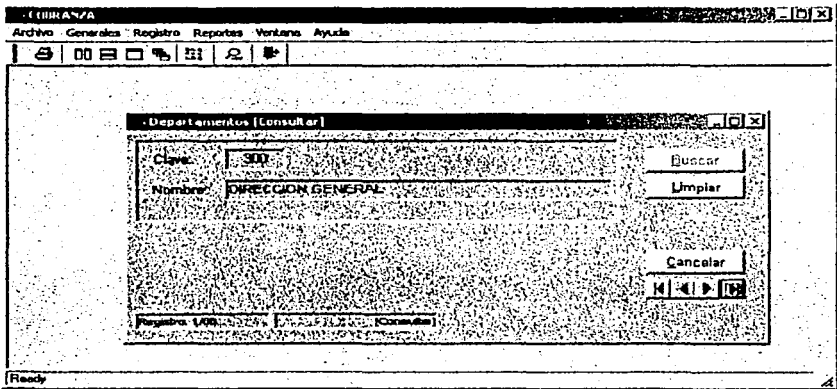


Figura A.23 Búsqueda registro por registro

### Eliminar un registro de un catálogo.

Para eliminar un registro debe dar click con el botón izquierdo del ratón en la opción "Eliminar". Así accederá a la ventana Eliminar del catálogo deseado. Observe la figura A.15.

Básicamente la ventana Eliminar presenta dos opciones **Buscar** y **Limpiar**; la primera es necesaria ya que primero que nada habrá que buscar el registro a borrar para su posterior eliminación.

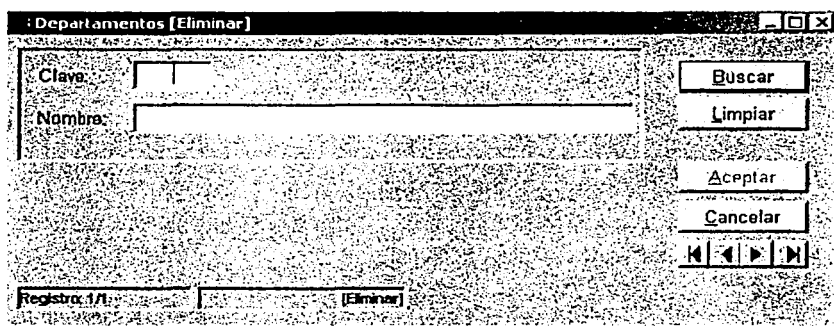


Figura A.24 Ventana "Eliminar" del catálogo Departamentos

Cuando el sistema devuelve los resultados de la búsqueda es necesario pulsar, con la ayuda del ratón, la opción "Aceptar" para eliminar el registro; de esta forma, aparecerá un aviso preguntando si se desea borrar el registro seleccionado, figura A.25, siempre y cuando usted esté seguro de borrar ese registro deberá oprimir "Si".

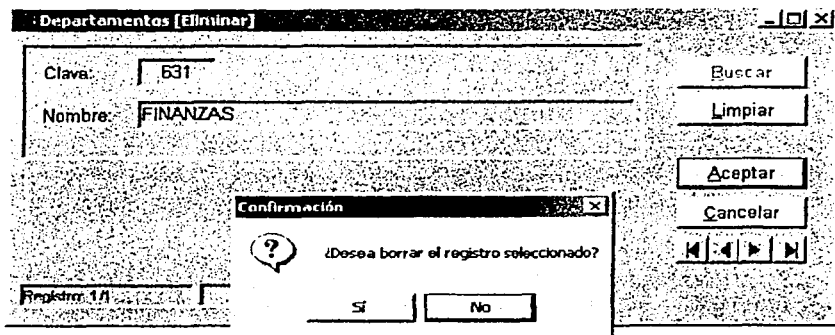


Figura A.26 Aviso de confirmación para borrar el registro seleccionado

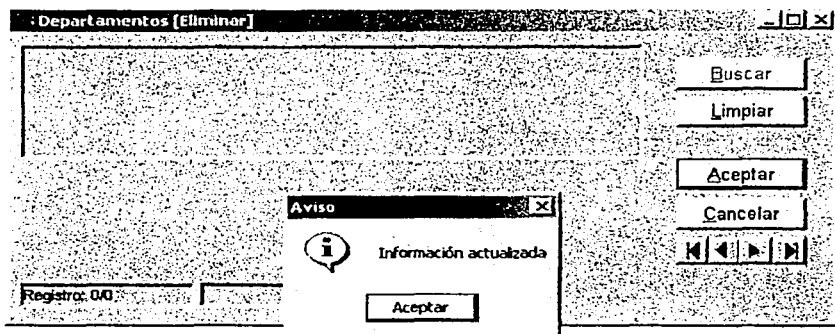


Figura A.26 Aviso de información actualizada

De lo contrario oprima "No" y en seguida "Limpiar" para iniciar otra búsqueda y encontrar el registro deseado.

Finalmente seleccione "Cancelar" para volver a la pantalla principal del catálogo.

## Cómo modificar los permisos de un usuario.

Otra opción contenida en esta sección es "Accesos Usuario", la cual tiene como finalidad habilitar permisos a los usuarios para acceder a ciertas o todas las opciones del menú dependiendo del departamento. De esta forma, en la figuras A.27 y A.28 se presentan la elección de la opción y la ventana de trabajo de esta, respectivamente.

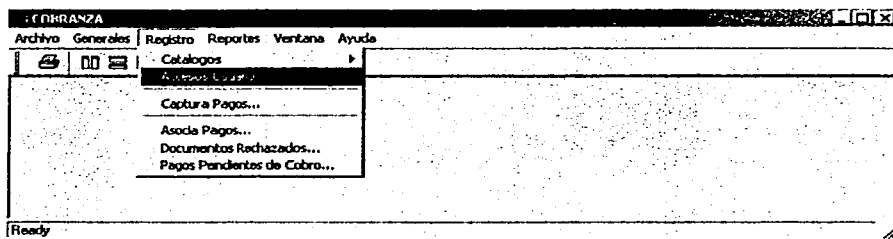


Figura A.27 Opción Accesos Usuarios

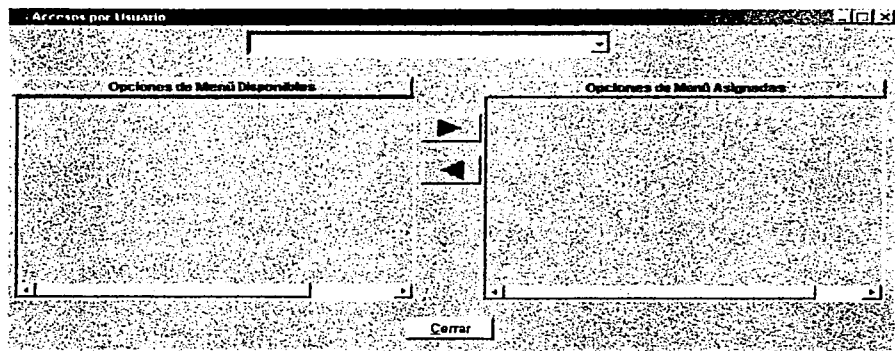


Figura A.28 Ventana de la Opción Accesos Usuarios

En la figura A.28 se puede observar que la ventana cuenta con un campo combo en el que se listan todos los usuarios. Por ello para habilitar o inhabilitar los menús a los usuarios se tendrá primero que el elegir al usuario deseado.

A continuación, se presentan dos ventanas: en una de ellas se muestra las opciones que el usuario ya tiene "asignadas" y en la otra las que puede asignárselle, o en caso contrario inhabilitarle. Cabe señalar que esta función solo puede ejecutarla el administrador o en su defecto el jefe del departamento.

Veamos el ejemplo siguiente, figura A.29. En este caso el administrador ha elegido a la usuario Azucena, la cual ya tiene algunas opciones del Menú asignadas pero se requiere que ella tenga acceso a la opción **Asocia Pago** del menú **Registro**. De esta forma, primero se elige esta opción de la ventana izquierda y se oprime el botón con la flecha derecha para autorizarle esta opción. En la figura A.30 se muestra el resultado: el usuario ya tiene este permiso asignado, ver ventana derecha.

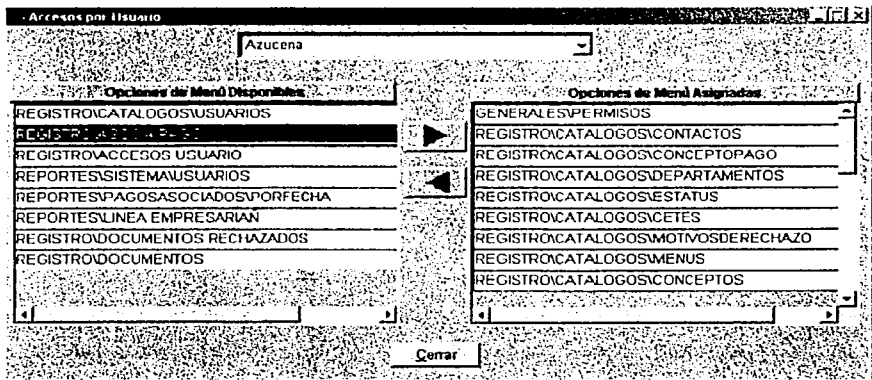


Figura A.29 Ejemplo de asignación de la opción Registro/Asocia Pago

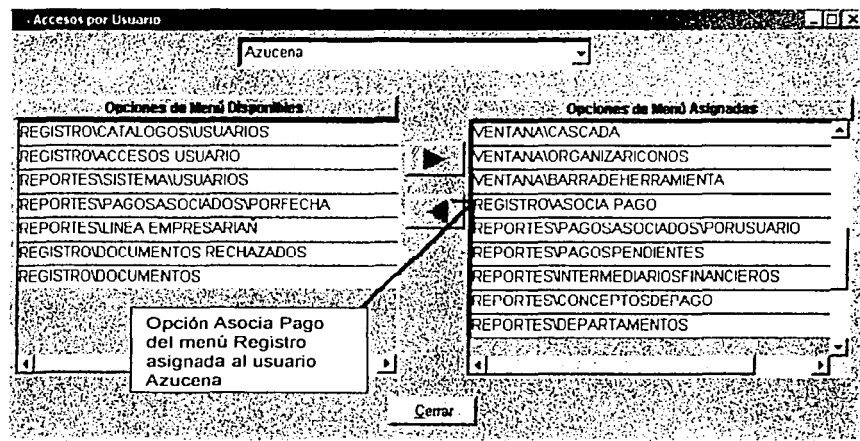


Figura A.30 Asignación exitosa de la opción Asocia Pago del menú Registro

- La segunda sección contiene la opción: **Captura Pagos**.

Como su nombre lo indica esta opción le permitirá capturar los pagos de los intermediarios financieros, así como realizar búsquedas de registros. La ventana cuenta con cuatro botones:

- **Buscar:** Como su nombre lo indica le ayudará a buscar el registro deseado.
- **Limpiar:** Limpia la pantalla para realizar nuevas búsquedas.
- **Nuevo:** Para ejecutar la captura de un nuevo pago
- **Cerrar:** Nos permite cancelar la operación para regresar a la pantalla principal.

También se visualizan los campos necesarios en la captura de los pagos o incluso en la búsqueda de registros, figura A.31:

- Folio
- Tipo de Pago
- Concepto Pago
- Intermediario Financiero
- Monto de Pago
- Fecha Recepción
- Fecha Documento
- Status

The screenshot shows a software window titled "Captura de Pagos de Intermediarios Financieros". The window contains a form with the following fields and controls:

- Folio:** A text input field.
- Tipo Pago:** A dropdown menu.
- Concepto Pago:** A dropdown menu.
- Intermediario Financiero:** A dropdown menu.
- Monto Pago:** A text input field.
- Fecha Recepción:** A text input field.
- Fecha Documento:** A text input field.
- Status:** A dropdown menu.

On the right side of the form, there are four buttons: "Buscar", "Limpiar", "Nuevo", and "Cerrar". At the bottom left of the window, it displays "Registros: 00" and a set of navigation arrows (back, forward, search, etc.).

Figura A.31 Ventana de Captura de pagos de Intermediarios Financieros

### Buscar y Limpiar:

En lo que respecta a las consultas, estas se realizan como en los catálogos, por ejemplo podemos teclear el número de folio y oprimir el botón "Buscar", con la ayuda del mouse. Entonces el sistema devolverá toda la información, siempre que exista el registro con el dato tecleado. Para realizar una nueva búsqueda solo deberá pulsar limpiar, así todos los campos estarán en blanco.

**Nuevo:**

Para acceder a la ventana de la opción Nuevo tendrá que oprimir el botón Nuevo, y a continuación se muestra la ventana, ver figura A.32 con los botones Aceptar, en caso que todos los campos se hayan capturado o Cancelar, para terminar y volver al menú principal de Captura de Pagos de Intermediarios Financieros.

The screenshot shows a software window titled "Captura de Pagos de Intermediarios Financieros". The window contains a form with the following fields and controls:

- Nombre:** A text input field.
- Tipo Pago:** A dropdown menu.
- Concepto Pago:** A dropdown menu.
- Intermediario Financiero:** A dropdown menu.
- Monto Pago:** A text input field.
- Fecha Recepción:** A text input field.
- Fecha Documento:** A text input field.
- Status:** A dropdown menu.
- Buttons:** "Aceptar" and "Cancelar" buttons are located on the right side of the form.
- Footer:** At the bottom left, it displays "Registros: 0/0" and a set of navigation arrows (back, forward, etc.).

**Figura A.32 Ventana de Captura de un nuevo Pago**

Al realizar la captura de los campos mostrados en la pantalla es necesario no omitir ninguno; pues el sistema nos mostrará un mensaje como el desplegado en la figura A.33, y deberá llenarlo antes de continuar, ya que el sistema no permitirá avanzar hasta que se haya capturado.



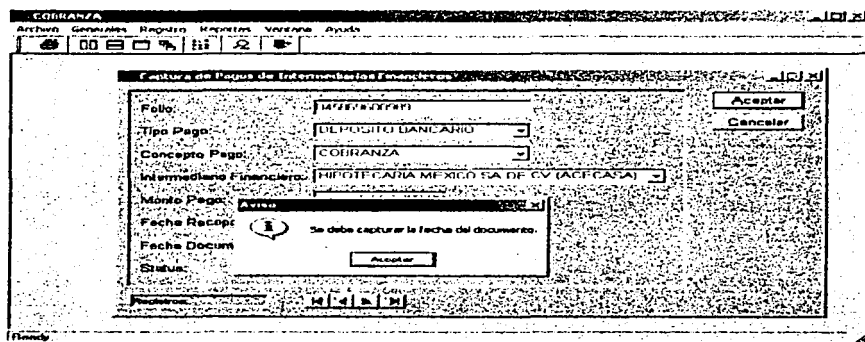


Figura A.33 Aviso de omisión del campo Fecha del documento

Es importante señalar, que la fecha de recepción no debe ser mayor a la del día actual, figura A.34, y que la fecha del documento tampoco debe exceder a la de la fecha de recepción del documento; por lo anterior, el sistema tomará por default la fecha del sistema y no permitirá ningún cambio en el campo **Fecha de Recepción** y si llegará a introducir una fecha mayor en el campo **Fecha Documento** el sistema enviará un aviso indicando esta situación para su corrección inmediata, figura A.35.

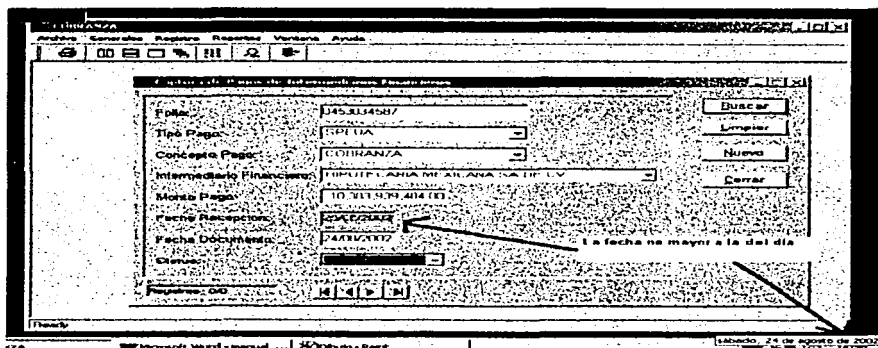


Figura A.34 Fecha Recepción no debe ser mayor a la del día actual

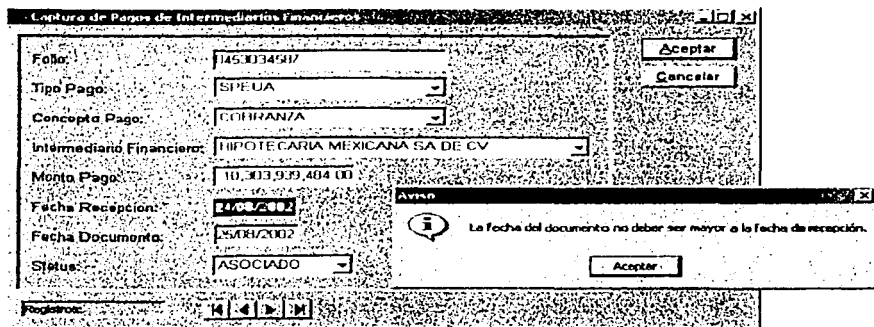


Figura A.35 Aviso de error al teclear la fecha de recepción

También tiene que tomarse en cuenta que al registrarse un nuevo pago el **Status** del mismo tiene que ser Activo, pues aún no se le ha asociado a ningún cobro pendiente.

Una vez capturados todos los campos y teniendo en cuenta todas las consideraciones anteriores, se debe pulsar **Aceptar** para salvar el nuevo registro y el sistema le mostrará el aviso: **Se dieron de alta correctamente los datos del documento**, figura A.36.

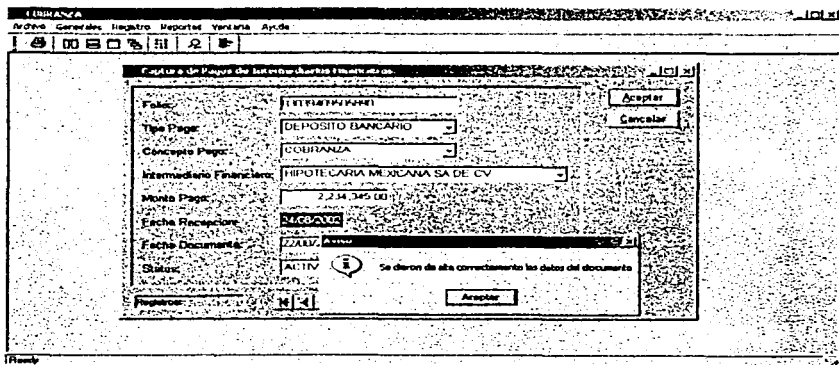


Figura A.36 Aviso de alta exitosa

- La tercera se enfoca a asociar los pagos, registrar los documentos rechazados y actualizar los pagos pendientes de cobro.

### Asociación de Pagos.

Esta opción tiene por objeto relacionar los pagos capturados con los abonos que se registraron en la cuenta empresarial. En la figura A.37 puede observarse que la ventana de esta opción se divide en dos recuadros: el primero de ellos muestra todos los pagos que faltan por asociar y el segundo los movimientos obtenidos de la línea empresarial.

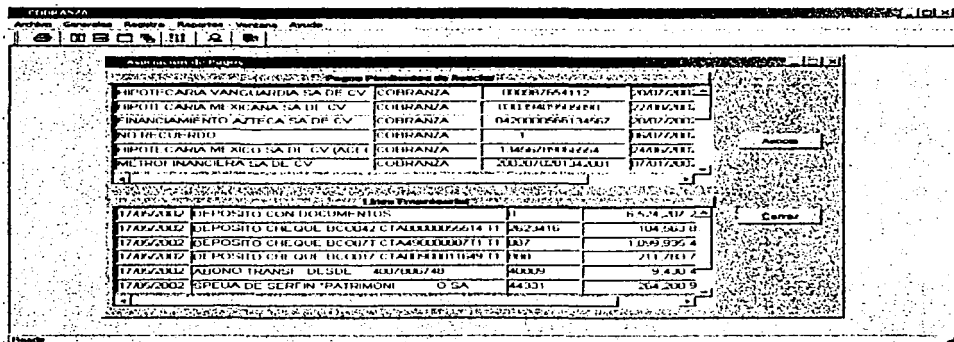


Figura A.37 Ventana de Asociación de Pagos

Para asociar los pagos sólo tiene que seleccionar el pago pendiente por asociar con la ayuda del mouse, coloque el apuntador del mouse sobre el pago deseado en **Pagos pendientes por asociar** y oprima el botón derecho para seleccionarlo, repita la operación en la **ventana Línea Empresarial**, figura A.38. Posteriormente, oprima **Aceptar** y si coinciden los montos automáticamente desaparecerán de la pantalla ambos registros. De lo contrario el sistema enviará un mensaje informando que los montos no corresponden figura A.39.

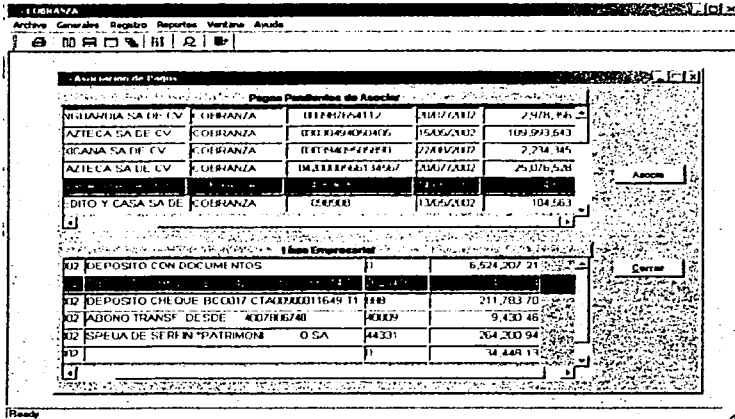


Figura A.38 Selección de los pagos a relacionar

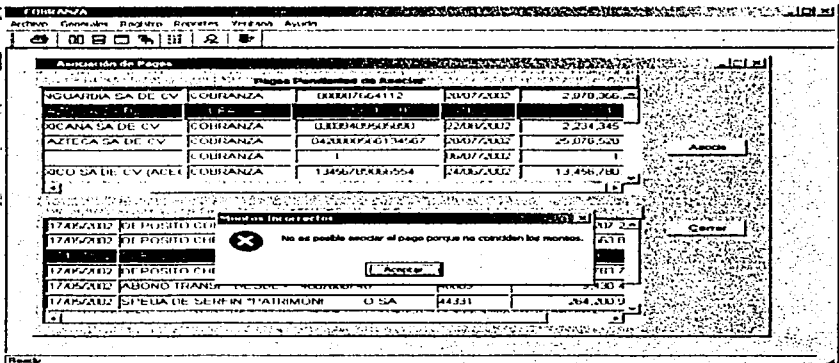


Figura A.39 Aviso de montos incorrectos al asociar pagos erróneos

Documentos Rechazados

El sistema cuenta con la opción de registrar todos aquellos pagos con documentos que hayan sido rechazados, ya sea por falta de fondos o cualquier otro motivo. La figura A.40 muestra la pantalla que resulta de dicha selección. La ventana consta de:

- un recuadro en donde se listan los pagos pendiente de asociar
- Botón Registrar
- Botón Cancelar

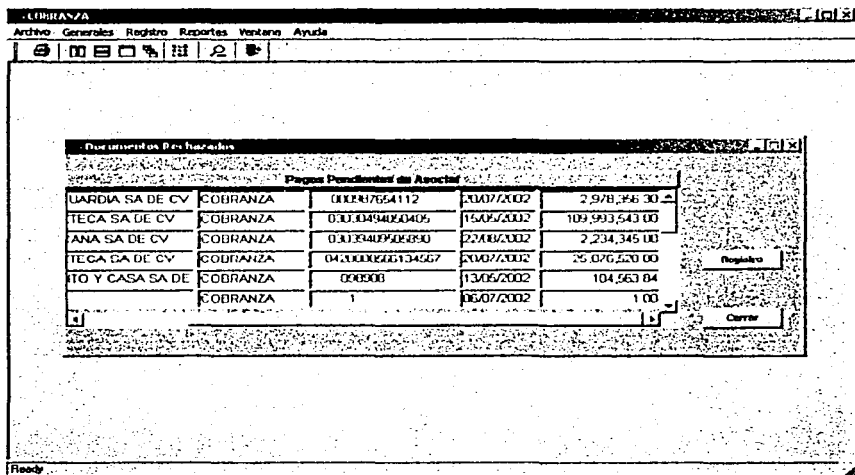


Figura A.40 Ventana de Documentos Rechazados

El registro del documento rechazado se efectua de la siguiente forma:

- Primero, tiene que elegir el pago rechazado con la ayuda del mouse.
- Después de verificar es necesario oprimir el botón Registro para salvar la operación.
- Si usted ya no requiere realizar otra operación en esta ventana, oprima el botón cancelar para volver a la pantalla principal.

## Pagos Pendientes de Cobro

El sistema también cuenta con la opción Pagos pendiente de cobro, la cual consiste en actualizar la lista de los pagos registrados después de efectuar la operación de asociación de pagos o del registro de documentos rechazados.

La ventana resultante de esta selección esta compuesta de la siguiente forma:

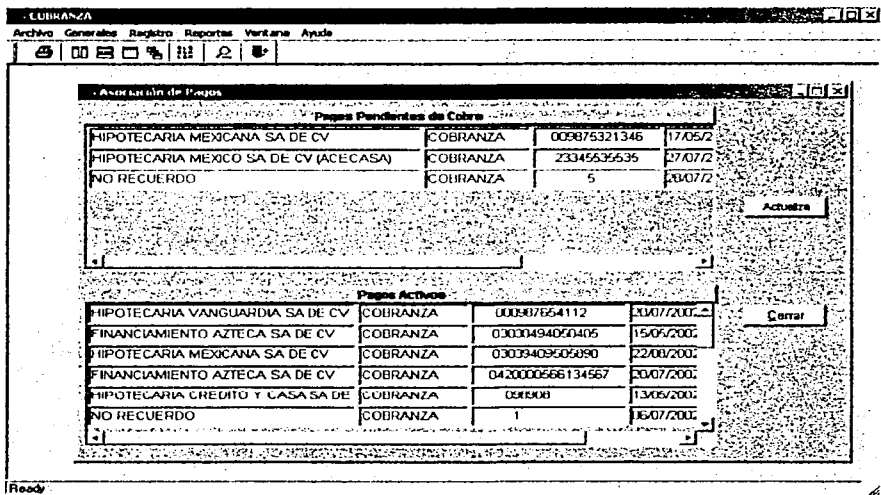


Figura A.41 Pagos Pendientes de Cobro

- un recuadro de Pagos Pendientes de Cobro
- una pequeña ventana en la que se listan los Pagos Activos
- Botón Registrar
- Botón Cancelar

Para realizar esta operación usted tiene que elegir de la lista de **Pagos Activos** aquél que sigue pendiente de cobro y posteriormente pulsar **Actualizar** para salvar la operación.

### Reportes.

En esta sección el sistema proporciona los reportes de:

Pagos Asociados (por usuario y por fecha), Pagos Pendientes, Intermediarios Financieros, Conceptos de Pago, Departamentos, Estatus de Proceso, Tasa Cetes Vigentes, Tipos de Rechazo, Tipos de Pago, Línea Empresarial y Sistema (Menús, Usuarios y Permisos).

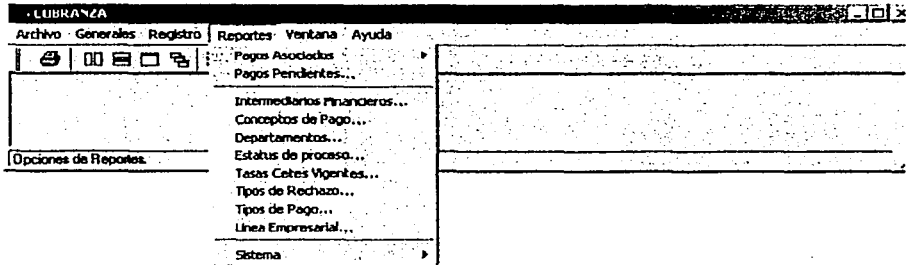


Figura A.42 Menú Reportes

En general cada una de las ventanas cuenta con cuatro botones:

- Configurar impresora
- Impresión preliminar
- Imprimir
- Cerrar

En lo que se refiere al botón **Configurar Impr.** la figura A.43 nos muestra la ventana resultante donde se podrá elegir la impresora deseada u oprimir **Setup** para configurar una nueva impresora. Al pulsar el botón de **impresión preliminar** el sistema abrirá la ventana en donde se observará la vista preliminar de impresión del reporte con la opción de ampliarla o reducirla; además de habilitar la opción de ver la regla, véase la figura A.44.

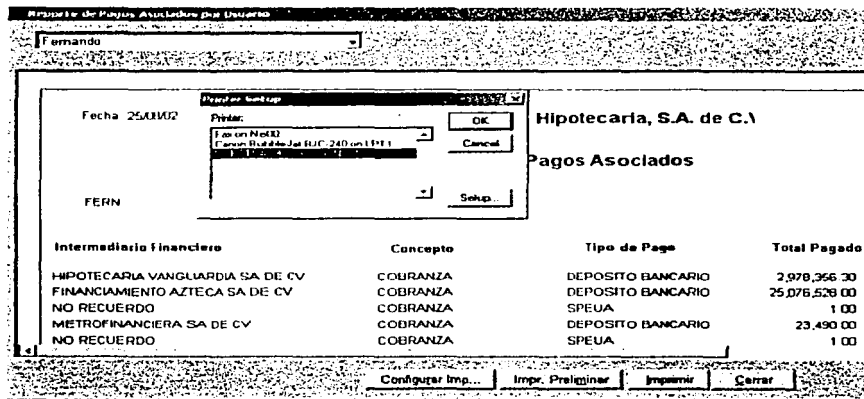


Figura A.43 Configurar Impresora



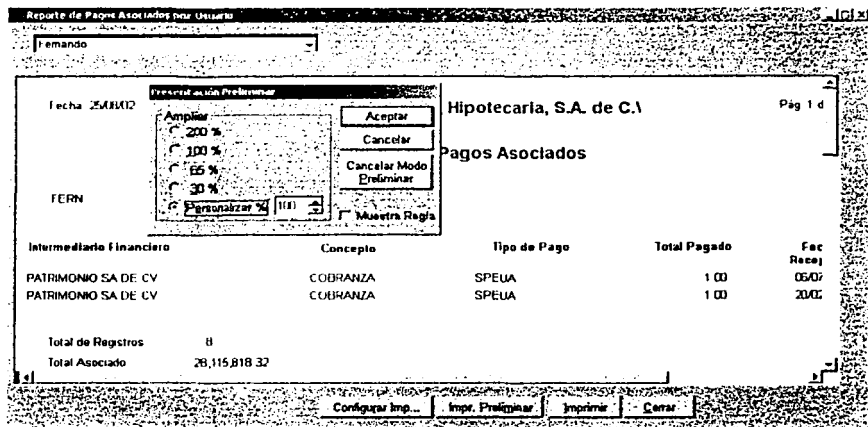


Figura A.44 Impresión preliminar

A continuación se tratará cada una de las opciones de reportes.

### Pagos Asociados

Este reporte nos presenta todos aquellos pagos que ya han sido asociados a los depósitos en la línea empresarial y es posible visualizarlos a través de la fecha o el nombre del usuario que efectuó dicha operación.

Además, los reportes de Pagos Asociados, ya sea por usuario o fecha, cuentan precisamente con un campo combo donde se listan los nombres los usuarios y las fechas, respectivamente, para así elegir el más indicado, figura A.45 y A.46.

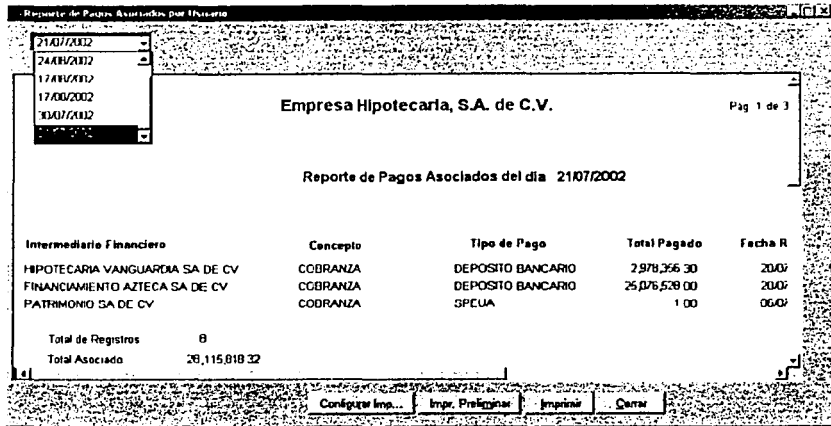


Figura A.45 Reporte de Pagos Asociados por Fecha

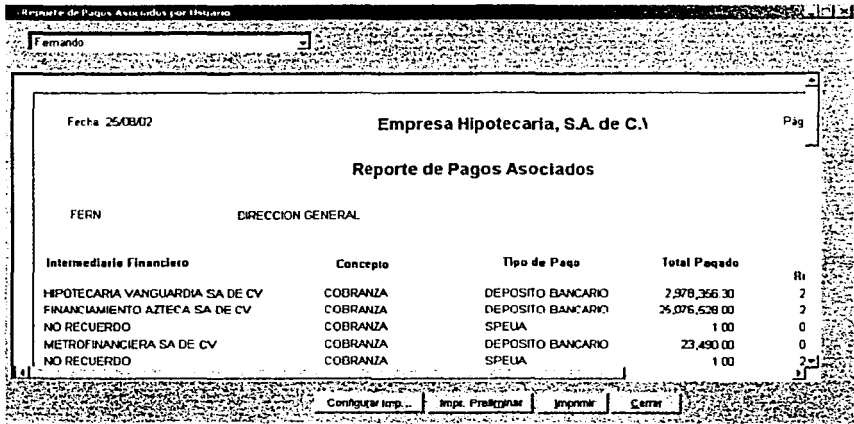


Figura A.46 Reporte de Pagos Asociados por Usuario

## Pagos Pendientes.

El reporte pagos pendiente mostrará todos los pagos capturados en el submenú **Registro** que aún no han sido asociados a los abonos registrados en la línea empresarial. En la figura A.47 se muestra la ventana resultante y cómo podrá observar la información se presenta por columnas indicando el intermediario financiero, tipo de pago, concepto, fecha de recepción, monto e intereses generados entre otros.

### Intermediarios Financieros.

En este reporte se nos presentan los contactos de cada uno de los Intermediarios Financieros registrados en la empresa hipotecaria.

Intermediario	Tipo de Pago	Concepto	Fecha de Recepción	Pagos Monto	Intereses Generados	Ad
HIPOTECARIA VANGUARDIA SA DE CV	DEPOSITO BANCARIO	COBRANZA	20/07/2002	2,978,366.30	32166.24	2
HIPOTECARIA MEXICANA SA DE CV	DEPOSITO BANCARIO	COBRANZA	17/05/2002	104,563.84	3136.92	
FINANCIAMIENTO AZTECA SA DE CV	DEPOSITO BANCARIO	COBRANZA	16/05/2002	109,993,543.00	333,2004.35	11L
HIPOTECARIA MEXICANA SA DE CV	DEPOSITO BANCARIO	COBRANZA	24/08/2002	2,234,345.00	670.3	2
FINANCIAMIENTO AZTECA SA DE CV	DEPOSITO BANCARIO	COBRANZA	20/07/2002	25,076,528.00	270826.5	2E
HIPOTECARIA VANGUARDIA SA DE CV	DEPOSITO BANCARIO	COBRANZA	16/05/2002	104,563.84	3168.29	

REPORTES/PAGOS

Configurar Imp... | Impr. Predefinir | Imprimir | Cerrar

Figura A.47 Reporte de Pagos Pendientes

### Conceptos de Pago

En la figura A.48 se presenta el reporte de los **conceptos de pago** definidos para este sistema, los cuales fueron establecidos por la Empresa Hipotecaria. La información que se muestra en éste es la clave del concepto y la descripción del mismo.

## Departamentos.

En lo que se refiere al reporte de **Departamentos** se presentará en éste la clave del departamento así como su descripción, figura A.49.

## Estatus de Proceso.

Todos los estatus posibles en el proceso de cobranza son considerados en este reporte, que al igual que los dos anteriores la información se distribuye en dos columnas: clave de estatus y descripción, observe la figura A.50.

Clave	Descripción
1	COBRANZA
2	INTERESES
3	PENALIZACION
4	CUOTA INGRESO
5	REACTIVACION

Fecha: 25/08/02      Empresa Hipotecaria, S.A. de C.V.      Pág 1 de 1

Reporte de Conceptos de Pago

Total de Registros: 5

Configurar Imp... | Imp. | Preservar | Imprimir | Cerrar

Figura A.48 Reporte de Conceptos de Pago

Reporte de Departamentos

Fecha 25/03/02 Empresa Hipotecaria, S.A. de C.V. Pág 1 de 12

**Reporte de Departamentos**

Clave Departamento	Nombre del Departamento
000	DIRECCION GENERAL
001	SECRETARIO DE CONSEJO
017	DIRECCION DE RELACIONES INSTITUCIONALES Y DE COMUN
000	SUBDIRECCION DE PROMOCION
004	DELEGACION DISTRITO FEDERAL
006	DELEGACION GUADALAJARA
006	DELEGACION CANCUN

Total de Registros 83

Configurar Imp... Impr. Predefinir Imprimir Correr

Figura A.49 Reporte de Departamentos

Reporte de Estatus

Fecha 25/03/02 Empresa Hipotecaria, S.A. de C.V. Pág 1 de 1

**Reporte de Estatus de Proceso**

Clave	Descripción
AC	ACTIVO
AS	ASOCIADO
AU	AUDITADO
BA	BAJA
CA	CANCELADO
PF	PFINIENTE
TE	TERMINADO

Total de Registros 7

Configurar Imp... Impr. Predefinir Imprimir Correr

Figura A.60 Reporte de Estatus de Proceso

## Tasa Cotes Vigentes

El reporte de Tasas de CETES vigentes nos muestra la fecha publicada y el valor de la tasa de CETES que se han utilizado para el cálculo de los intereses moratorios, figura A.51.

### Tipos de Rechazo.

Todos los motivos de rechazo de los pagos recibidos durante el proceso de la cobranza son desplegados en este reporte, el cual presenta en dos columnas la clave del rechazo y su descripción, figura A.52.

Fecha Publicada	Valor
15/06/2002	14.25
22/06/2002	13.96
29/06/2002	13.56
14/07/2002	1.20
29/07/2002	10.56
05/08/2002	10.00
10/08/2002	6.50

Figura A.51 Reporte de Estatus de Proceso

Reporte de Motivos de Rechazo en Pagos Recibidos

Fecha: 25/08/02      Empresa Hipotecaria, S.A. de C.V.      Pag: 1 de 1

### Reporte de Motivos de Rechazo de Pagos Recibidos

Clave	Descripción
1	CHEQUE SIN FONDOS
2	CUENTA CANCELADA
3	CUENTA DEFUNCTA
4	Inconsistencias en el Pago

Total de Registros: 4

Configurar Imp...    Imp. Preliminar    Imprimir    Cerrar

Figura A.62 Reporte de Tipos de Rechazo

### Tipos de Pago

Al igual que el reporte anterior se presenta a dos columnas la clave del tipo de pago y su descripción. Sólo se consideran como tipos de pagos válidos los realizados por SPEUA, depósito bancario, cheque y efectivo en caja, figura A.53.

Reporte de Tipos de Pagos Recibidos

Fecha: 25/08/02      Empresa Hipotecaria, S.A. de C.V.      Pag: 1 de 1

### Reporte de Tipos de Pagos Recibidos

Clave	Descripción
1	SPEUA
2	DEPOSITO BANCARIO
3	CHEQUE
4	EFFECTIVO EN CAJA

Total de Registros: 4

Configurar Imp...    Imp. Preliminar    Imprimir    Cerrar

Figura A.63 Reporte de Tipos de Pago

## Línea Empresarial.

Esta opción de reportes nos permitirá verificar todos los movimientos registrados en la cuenta bancaria de la Empresa Hipotecaria. Recuerde que esta información se generó a partir de una conexión directa con el banco que maneja las cuentas de la Empresa Hipotecaria (submenú **Generales \ Línea empresarial**), observe la figura A.54.

Reporte de Pagos de Línea Empresarial No Asociada a Cuenta

Fecha: 25/08/02 Empresa Hipotecaria, S.A. de C.V. Pag: 1 de 2

Línea Empresarial pendiente de Asociar del día 02/08/2002

fecha Estado de Cuenta	Movimiento	Descripción	Monto Abonado	Sucursal	Banco Origen
02/08/2002	0	ABONO POR COBRANZA	7294.6300	3005	000
02/08/2002	0	ABONO POR COBRANZA	24187.7900	3005	000
02/08/2002	0	ABONO POR COBRANZA	4982.5000	520	000
02/08/2002	3	ABONO POR COBRANZA	3000.4500	632	000
02/08/2002	0	ABONO POR COBRANZA	7821.5000	632	000
02/08/2002	0	ABONO POR COBRANZA	17072.7700	632	000
02/08/2002	0	ABONO POR COBRANZA	10666.6600	749	000
02/08/2002	0	ABONO POR COBRANZA	3900.6000	201	000

Configurar Imp... Impr. Preeliminar Imprimir Cerrar

Figura A.54 Reporte de Tipos de Pago

## Sistema (Menús, Usuarios y Permisos).

Los reportes de la opción Sistema incluyen tanto los de los menús disponibles (figura A.55), los usuarios dados de alta (figura A.56) y los permisos otorgados a cada uno de



ellos sobre los menús(figura A.57). Cabe señalar que este tipo de reportes es manejado por el administrador del sistema para llevar un control más estricto de éste.

Reporte de Menús del Sistema

Fecha 25/07/02 Empresa Hipotecaria, S.A. de C.V. Pag 1 de 6

### Reporte de Menús del Sistema

Opcion	Ruta del Menu	Nombre del Menu
1	GENERALES/PERMISOS	m_permisos
2	REGISTROCATALOGOS/CONTACTOS	m_contactos
3	REGISTROCATALOGOS/CONCEPTOPAGO	m_conceptopago
4	REGISTROCATALOGOS/DEPARTAMENTOS	m_departamento
5	REGISTROCATALOGOS/ESTATUS	m_estatus
6	REGISTROCATALOGOS/ETES	m_cetas
7	REGISTROCATALOGOS/USUARIOS	m_usuarios

Total de Registros: 36

Configurar Imp... Impr. Práctimase Imprimir Cerrar

Figura A.55 Reporte de Menús del Sistema

Reporte de Usuarios del Sistema

Fecha 25/07/02 Empresa Hipotecaria, S.A. de C.V. Pag 1 de 1

### Reporte de Usuarios del Sistema

Nombre	Clave Usuario	Departamento	Status
ADMINISTRATOR	ADMINIST	DIRECCION DE TECNOLOGIA DE INFORMACION	ACTIVO
Alejandro	ALEX	SECRETARIO DE CONSEJO	ACTIVO
Proyecto Tesis	AZKABAN	SUBDIRECCION DE COMPUTO	ACTIVO
Azucena	AZUC	DELEGACION TLUANA	ACTIVO
Fernando	FERN	DIRECCION GENERAL	ACTIVO
Josefina Cruz alonso	Josefina	SUBDIRECCION DE COMPUTO	ACTIVO
NIEVES GARCIA	NIEVES	DIRECCION DE TECNOLOGIA DE INFORMACION	ACTIVO
TESIS	TESIS	DIRECCION DE TECNOLOGIA DE INFORMACION	ACTIVO

Configurar Imp... Impr. Práctimase Imprimir Cerrar

Figura A.56 Reporte de Usuarios del Sistema

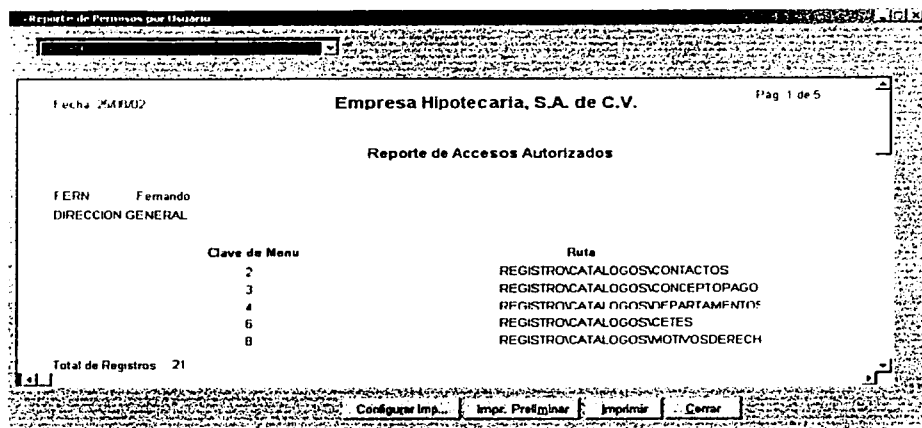


Figura A.57 Reporte de Accesos Autorizados

### Ventanas.

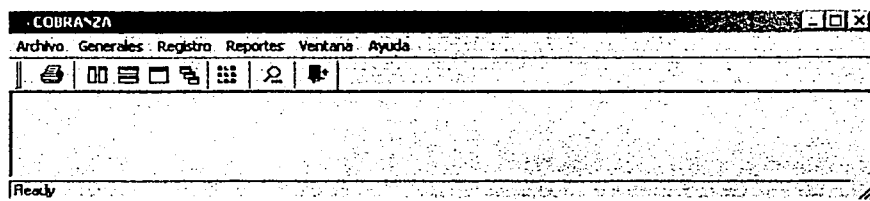


Figura A.58 Submenú Ventana

En sección comprende todos los tipos de despliegue de las ventanas ya sea en forma: Vertical, Horizontal, Mosaico, Cascada. Asi como Organizar Iconos y Barra de Herramientas.

## Ayuda

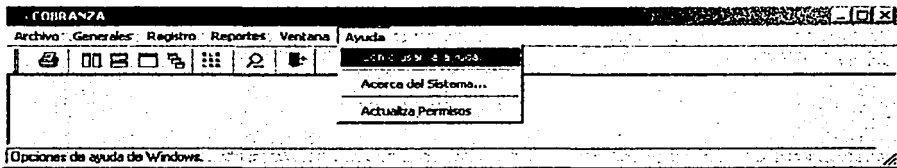


Figura A.59 Submenú Ayuda

Ésta contiene :

- > Como Usar la Ayuda
- > Acerca del sistema.
- > Actualizar Permisos.

Respecto a la opción **Cómo usar la Ayuda**, se nos presenta la Ayuda de Microsoft Windows, veamos la figura A.60, la cual es muy familiar a todos los usuarios de Windows pues presenta una pequeña ventana donde el usuario podrá buscar o resolver sus dudas a través del índice o la búsqueda por palabras, figura A.61.

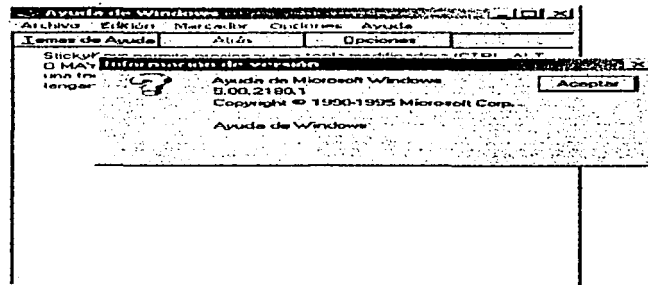


Figura A.60 Ayuda de Windows

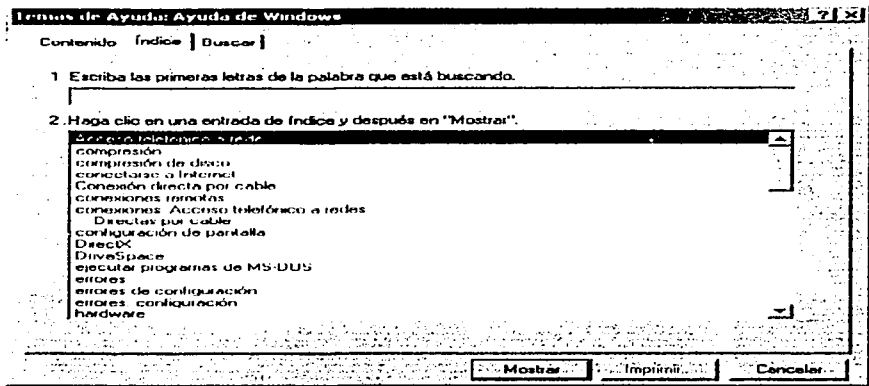


Figura A.61 Ventana de Temas de Ayuda

En la opción Acerca del Sistema se muestra la información más relevante de éste, tal es el caso del nombre del Sistema, la versión y la fecha, por mencionar algunos, ver figura A.62.

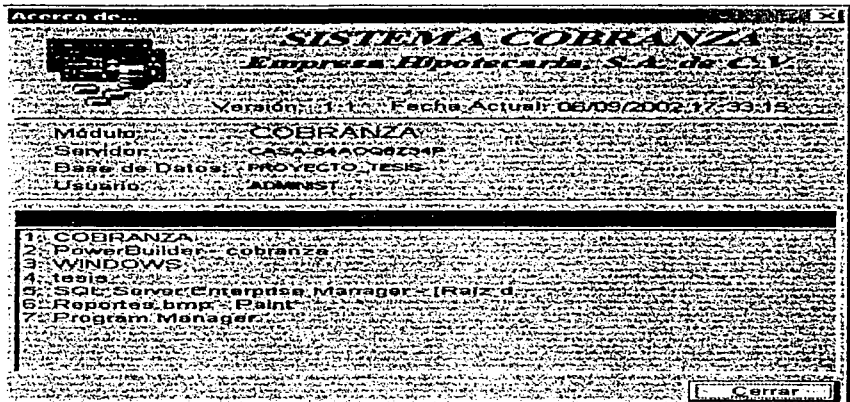


Figura A.62 Ventana Acerca del Sistema

Finalmente, la opción Actualiza Permisos nos permite, en caso de tener los privilegios sobre los accesos a las opciones del menú, actualizar los cambios realizados; solo basta con pulsar esta opción y el sistema desplegará un aviso cómo en la figura A.63 se muestra.

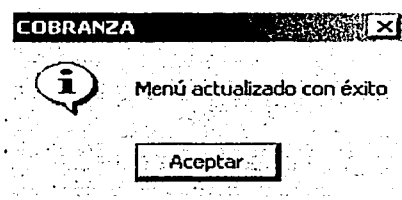


Figura A.63 Mensaje de Actualización del Menú

## MANUAL TÉCNICO

Para tener más información sobre la estructura del sistema, realizar tareas de mantenimiento y soporte técnico, es necesario proporcionar un manual técnico, el cual se presenta a continuación.

Este documento está compuesto de información sobre formularios y módulos que integran el sistema, además, los pasos a seguir para la instalación y personalización eficiente del sistema, así como las necesidades o requerimientos técnicos tanto del software como del hardware.

### Módulos que forman la aplicación:

<i>N°</i>	<i>Módulo</i>	<i>Descripción</i>
1	Cobranza	Este módulo contiene todas las librerías necesarias para el funcionamiento del sistema

### Catálogos que forman el sistema:

<i>N°</i>	<i>Catálogo</i>	<i>Descripción</i>
1	Cat_Motivo_Rechazo	Contiene el motivo por el que un pago fue rechazado.
2	Cat_Tip_pag	Incluye los diferentes tipos de pagos.
3	CETES	Contiene el valor del CETE actual.
4	Cta_Status	Contiene los estatus.
5	Departamento	Contiene el nombre de los departamentos.
6	Contactos_IF	Contiene el nombre de los Intermediarios Financieros.
7	Lineaempresarial	Contiene la carga de la línea empresarial
8	Usuario	Contiene los nombres de

## MANUAL TÉCNICO

Para tener más información sobre la estructura del sistema, realizar tareas de mantenimiento y soporte técnico, es necesario proporcionar un manual técnico, el cual se presenta a continuación.

Este documento está compuesto de información sobre formularios y módulos que integran el sistema, además, los pasos a seguir para la instalación y personalización eficiente del sistema, así como las necesidades o requerimientos técnicos tanto del software como del hardware.

### Módulos que forman la aplicación:

<i>N°</i>	<i>Módulo</i>	<i>Descripción</i>
1	Cobranza	Este módulo contiene todas las librerías necesarias para el funcionamiento del sistema

### Catálogos que forman el sistema:

<i>N°</i>	<i>Catálogo</i>	<i>Descripción</i>
1	Cat_Motivo_Rechazo	Contiene el motivo por el que un pago fue rechazado.
2	Cat_Tip_pag	Incluye los diferentes tipos de pagos.
3	CETES	Contiene el valor del CETE actual.
4	Cta_Status	Contiene los estatus.
5	Departamento	Contiene el nombre de los departamentos.
6	Contactos_IF	Contiene el nombre de los Intermediarios Financieros.
7	Líneaempresarial	Contiene la carga de la línea empresarial
8	Usuario	Contiene los nombres de

		usuario
9	Menus	Contiene los menús del sistema
10	Concepto_pago	Contiene el concepto de pago
11	DocxCobrar	Muestra los documentos pendientes de cobro.
12	Asocia_pago	Este catálogo asocia un pago.
13	UsuariosAcceso	Contiene la lista de los usuarios que tienen acceso al sistema
14	Pagos	Contiene el folio del documento que realizó un pago

### Funciones Globales del sistema:

N°	Función
1	Global funcion
2	Global menu
3	Global userobject
4	Global window

### Funciones del sistema:

N°	Función	Descripción
1	F_avanza_fecha	Avanza a la siguiente fecha
2	F_barrido_menu	Hace un barrido de menús
3	F_barrido_menu_permisos	Hace un barrido de los menús con permisos
4	F_centra_ventana	Centra la ventana de la aplicación
5	F_inicia_conexión	Inicia una conexión con una Base de Datos
6	F_logo	Inserta el logo Pagos.bmp
7	F_modifica_usuario	Modifica los accesos de usuarios
8	F_modos_permisos	Verifica los modos de permisos
9	F_obten_directorio_actual	Obtiene el directorio específico para un archivo
10	F_obten_elemento_menu_principal	Obtiene un elemento del menú principal.
11	F_obten_menu_principal	Obtiene el menú principal
12	F_obten_usuario	Obtiene el usuario
13	F_popmenu_ordenar	Ordena los menús
14	F_seleccion_multiple	Permite realizar selecciones



15	F_valida_cadena	múltiples Valida una cadena de caracteres
16	F_valida_identificacion	Valida la identificación de usuario
17	F_valida_permiso_menu	Valida los permisos de los menús
18	F_valida_usuario	Valida a los usuarios válidos

### Menús del sistema:

<i>Nº</i>	<i>Menú</i>	<i>Descripción</i>
1	Principal	Proporciona el acceso a cualquier tarea dentro de la aplicación

### Ventanas del sistema:

<i>Nº</i>	<i>Ventana</i>	<i>Descripción</i>
1	W asignación_usuario	Asigna usuario
2	W barra_contador	Muestra una barra contadora
3	W captura_pagos	Permite capturar un pago
4	W_cat_cetes	Permite ver el catálogo de CETES
5	W_cat_concepto_tipo_pago	Manipulación de el concepto de pago
6	W_cat_departamentos	Manipulación de los departamentos
7	W_cat_estatus	Manipulación de los estatus
8	W_cat_if	Manipulación del Intermediario financiero
9	W_cat_motivo_rechazo	Manipulación del motivo de rechazo
10	W_child_catalogo_padre	Ventana heredada del catálogo padre.
11	W_child_ventana	Ventana que nos permite justificar la ventana de inicio.
12	W_child_mantenimiento	Nos permite dar mantenimiento al sistema.
13	W_detalle_consultas	Permite consultar.
14	W_detalle_librerías	Nos permite ver el detalle de las librerías.

15	W_detalle_mantenimiento	Nos permite ver el detalle del mantenimiento del sistema.
16	W_detalle_objetos	Nos permite ver el detalle de objetos.
17	W_detalle_tablas	Nos permite ver el detalle de tablas.
18	W_detalle_ventanas	Nos permite ver el detalle de ventanas.
19	W_logo	Nos permite insertar el logotipo del sistema.
20	W_mantenimiento	Nos permite dar mantenimiento al sistema.
21	W_mantenimiento_motivo_salida	Nos permite dar mantenimiento al motivo de salida.
22	W_mantenimiento_si_asigna_usuario	Nos permite dar mantenimiento a los usuarios asignados.
23	W_modos_permisos	Manipulación del modo de permisos
24	W_pagos	Ventana para capturar un pago.
25	W_permiso_salida	Ventana que nos permite salir.
26	W_presentacion	Ventana de presentación.
27	W_principal	Ventana principal del sistema.
28	W_principal_padre	Ventana padre.
29	W_registra_salida_equipo	Registra la salida del equipo.
30	W_usuario_acceso	Permite capturar el acceso a los usuarios.
31	W_ventana_consulta	Permite consultar catálogos.
32	W_ventana_menu	Permite consultar menús.
33	W_ventana_objeto	Permite consultar objetos.

**Datwindows asociados:**

N°	<i>Datawindows</i>	<i>Descripción</i>
1	Dw captura pagos	Catálogo de captura de pagos.
2	Dw cat cetes	Catálogo de CETES.
3	Dw cat concepto tipo pago	Catálogo del tipo de pago.
4	Dw cat departamento	Catálogo de departamento.
5	Dw cat estatus	Catálogo de estatus.
6	Dw_cat_if	Catálogo del intermediario financiero.

7	Dw_cat_motivo_rechazo	Catálogo del motivo de rechazo.
8	Dw_depto	Catálogo de departamento.
9	Dw_dd_concepto_pago	Catálogo del concepto de pago.
10	Dw_dd_if	Datawindow del intermediario financiero.
11	Dw_dd_status	Datawindow del status
12	Dw_dd_tipo_pago	Datawindow del tipo pago.
13	D_acceso	Permite tener acceso.

## B.2 Pasos para cargar el Sistema Cobranza en el cliente

El primer paso consiste en generar la carpeta *Sistema Cobranza* y cargar las librerías de PowerBuilder necesarias para el funcionamiento del sistema que se listan a continuación:

<i>PBODB70.DLL</i>	<i>PBADD70.DLL</i>
<i>PBCGR70.DLL</i>	<i>PBTRA70.DLL</i>
<i>PBCMP70.DLL</i>	<i>PBCGB70.DLL</i>
<i>PBDEV70.DLL</i>	<i>PBCGC70.DLL</i>
<i>PBDPP70.DLL</i>	<i>PBCGL70.DLL</i>
<i>PBDTS70.DLL</i>	<i>PBCH70.DLL</i>
<i>PBDWO70.DLL</i>	<i>PBCPP70.DLL</i>
<i>PBDWP70.DLL</i>	<i>PBORC70.DLL</i>
<i>PBGEN70.DLL</i>	<i>PBPRX70.DLL</i>
<i>PBLIB70.DLL</i>	<i>PBWPB70.DLL</i>
<i>PBAPL70.DLL</i>	<i>PBOGR70.DLL</i>
<i>PBRCT70.DLL</i>	<i>PBCOM70.DLL</i>
<i>PBSCR70.DLL</i>	<i>PBJPX70.DLL</i>
<i>PBSQL70.DLL</i>	<i>PBDWR70.DLL</i>
<i>PBSYS70.DLL</i>	<i>PBJAG70.DLL</i>
<i>PBUDO70.DLL</i>	<i>PBDIR70.DLL</i>
<i>PBVM70.DLL</i>	<i>PBIN970.DLL</i>
<i>PBSYJ70.DLL</i>	<i>PBMSS70.DLL</i>
<i>PBSYC70.DLL</i>	<i>PBO8470.DLL</i>
<i>PBJAG70.DLL</i>	<i>PBSCC70.DLL</i>

Posteriormente se cargará en cada cliente el Sistema Cobranza (dentro de la misma carpeta), junto con las imágenes necesarias para su funcionamiento en la carpeta: *Imágenes*, verificando las siguientes condiciones:

- Espacio en disco requerido 100 Megabytes
- Memoria RAM de 64 MB
- Tarjeta de red PCI (Operación Dual 10 BASE T/100 BASE TX)
- Mouse
- Monitor SVGA color 15"
- Windows 95 o superior

### B.3 Configuración del cliente

El inicio de sesión deberá estar a cargo del Administrador. El primer paso para la configuración del cliente es la instalación de SQL Server con los atributos de cliente.



Figura B.1 Instalación de SQL Server 2000 Standard

Al encontrarse el sistema con Windows 95, 98 o 2000, inmediatamente sugiere la instalación como cliente:

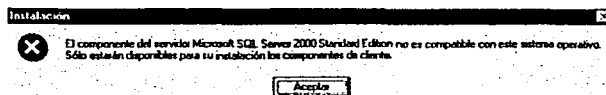


Figura B.2 Detección de una versión de Windows para cliente

Lo instalará en un equipo local y sólo las herramientas necesarias para el cliente.

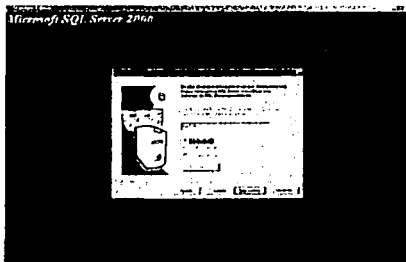


Figura B.3 Instalación en equipo local

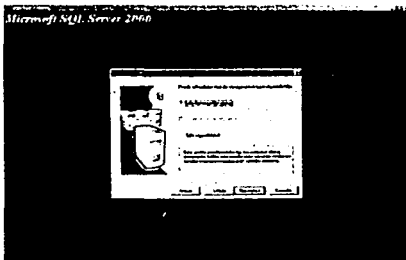


Figura B.4 Sólo herramientas de cliente

Posteriormente solicita el nombre del usuario y la compañía.

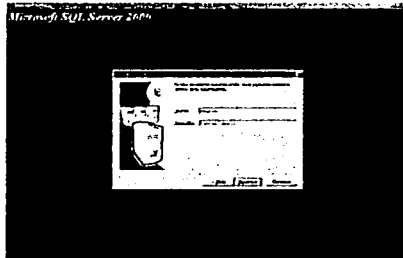


Figura B.5 Solicita el nombre de usuario y la contraseña

El siguiente paso consiste en reconocer el servidor, mediante el Administrador Corporativo, con los pasos como se listan en la siguiente figura:

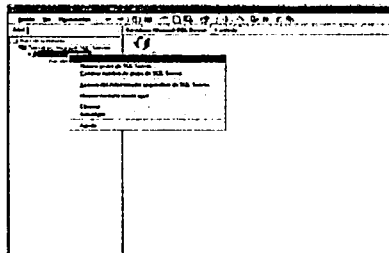


Figura B.6 Dar de alta el Servidor

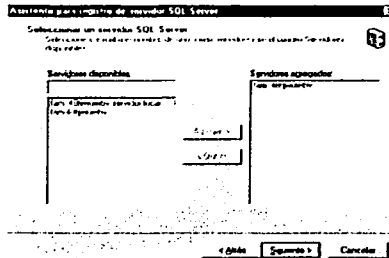


Figura B.7 Elección del servidor

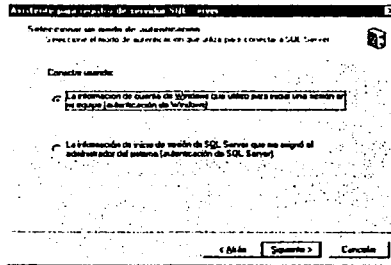


Figura B.8 Autenticación de Windows

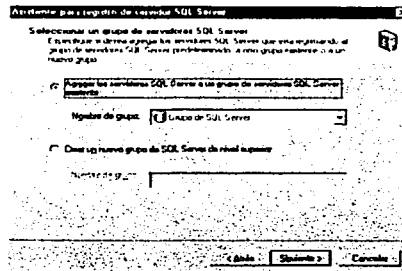


Figura B.9 Agregar los servidores a un grupo





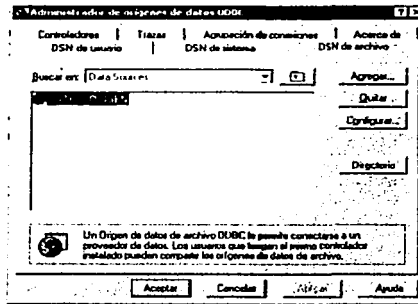


Figura B.12 DSN de archivo

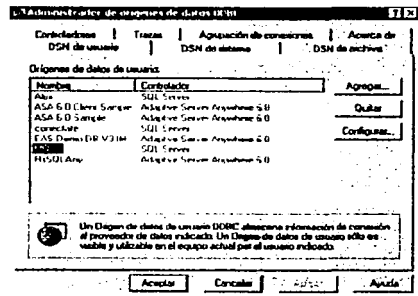


Figura B.13 DSN de Usuario

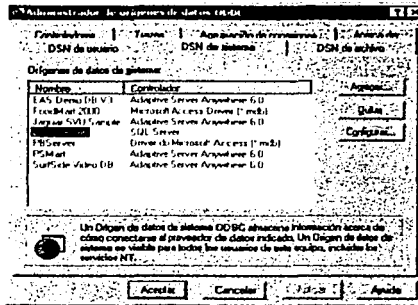


Figura B.14 DSN de Sistema

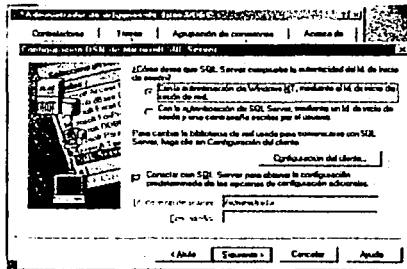


Figura B.15 Utilizando la autenticación de Windows

Es necesario que para establecer la conexión con el servidor y con la base de datos, el servidor esté activo:

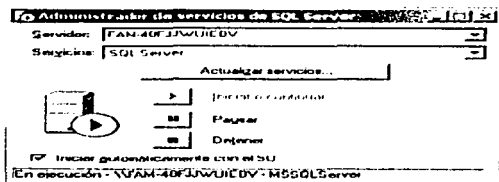


Figura B.16 Servidor Activo

Creación del nuevo origen de datos:

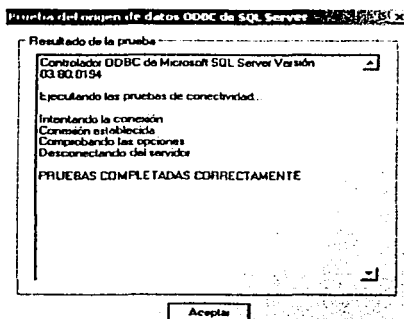


Figura B.17 Confirmación del origen de datos ODBC

Dicha configuración se realizará mediante el usuario de administrador, una vez concluida la configuración del ODBC.

### Configuración del servidor

El primer paso en la configuración del servidor consiste en dar de alta los usuarios correspondientes en Windows, dentro del servidor, con los perfiles correspondientes, en este caso, se dan de alta como invitados.

Una vez que ya están dados de alta en Windows, el siguiente paso consiste en crear claves de inicio de sesión mediante el Administrador Corporativo de SQL Server.

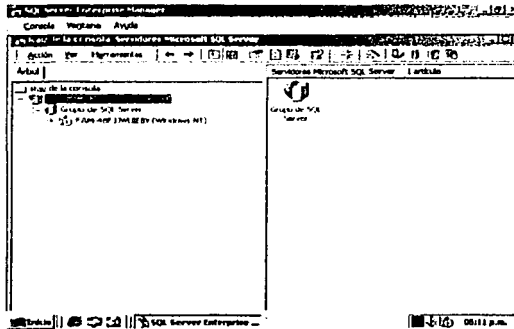


Figura B.18 Administrador Corporativo de SQL Server 2000

Para agregar una clave de inicio de sesión, se realiza mediante el Analizador de consultas de SQL Server:

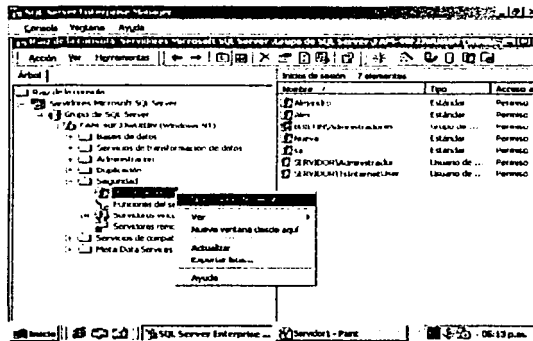


Figura B.19 Nuevo inicio de sesión

Dentro de las Funciones del Servidor, se especifican los privilegios de seguridad en todo el servidor:

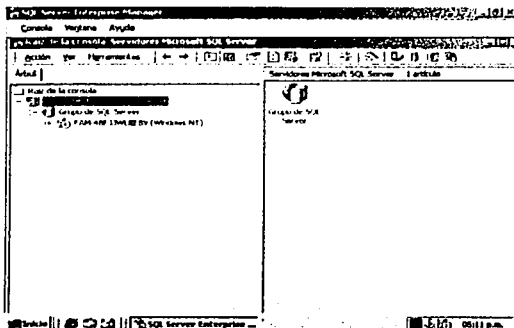


Figura B.18 Administrador Corporativo de SQL Server 2000

Para agregar una clave de inicio de sesión, se realiza mediante el Analizador de consultas de SQL Server:

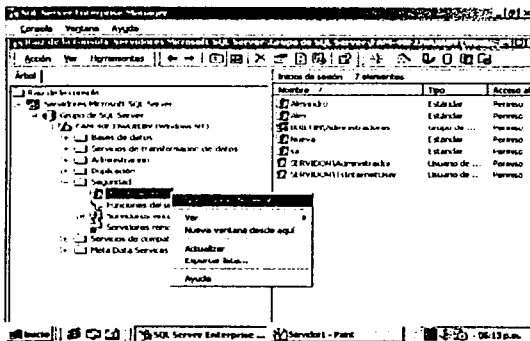


Figura B.19 Nuevo inicio de sesión

Dentro de las Funciones del Servidor, se especifican los privilegios de seguridad en todo el servidor:

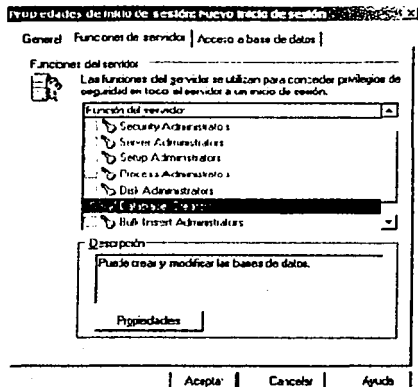


Figura B.20 Funciones del servidor

A continuación se le asocian las Bases de Datos a las que tendrán acceso:

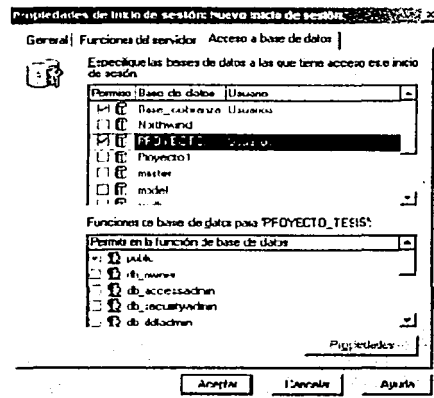


Figura B.21 Acceso a la Base de Datos



---

## BIBLIOGRAFÍA

- Anthony, Robert N.  
La contabilidad en la administración  
de empresas  
Unión Tipográfica  
México, Hispano – Americana, 1964
- Bolten, Steven E.  
Administración Financiera  
México, Limusa, 1983
- Brealey, Richard A. & Myers, Stewart  
C.  
Principios de Finanzas Corporativas  
4ª. edición  
España, McGraw-Hill, 1993
- Castaño, Adoración de Miguel &  
Plattini, Mario Gerardo  
Concepción y Diseño de Bases de  
Datos del modelo E/R  
México, Adisson Wesley, 1993
- Coffman, Gayle  
SQL Server 7 Referencia  
México, Osborne McGraw-Hill, 1999
- Elmasri, Ramez & Navathe, Sham  
Kant B.  
Sistemas de Bases de Datos  
Conceptos Fundamentales  
2ª. edición  
México, Pearson Educación, 2000
- Gallahger, Simón & Herbert, Simón  
J.A.  
Power Builder 4.0  
1a edición  
México, Sams Publishing, 1955
- Gar, Hanson W.  
Diseño y Administración de Bases de  
Datos  
McGrawHill
- Gilmore, W.J.  
A Programmer's Introduction to PHP  
4.0
- Gómez, Ricardo (Investigador CIFI-  
Informática Universidad de los Andes)  
DEA Informática  
Francia 1992
- Gramm, Ian  
Métodos Orientados a Objetos  
2ª. edición  
Adison/Wesley
- Guajardo Cantú, Gerardo  
Contabilidad Financiera  
2ª. edición  
México, McGraw-Hill, 1995
- Heys, William B.  
Power Builder 7.0.1 Edición especial  
1a edición  
México, Prentice may, 1988
- López-Fuensalida, Antonio  
Metodologías de Desarrollo,  
Producción Automática de Software  
con Herramientas Case  
Macrobit, 1991
- Martinez Espitia, Jaime Lorenzo;  
Perez Reyes, Luis &  
Romero Rodriguez, Guillermo  
Desarrollo de un sistema para la  
administración de hoteles y atención  
a clientes  
Tesis profesional de Ingeniería en  
Computación  
UNAM, 2002
-



Mendelzon, Ale Rubistein  
Introducción a las Bases de Datos  
Relacionales  
1ª. edición  
Buenos Aires, Prentice May, 2000

Presuman, Roger S.  
Ingeniería del software, un enfoque  
práctico  
México, McGraw-Hill, 1995.

Rivero, Cornelio E.  
Bases de Datos Relacionales  
2a. Edición  
España, Editorial Paraninfo, 1992

Secretaría de Hacienda y Crédito  
Público  
Banco de México (en su carácter de  
Agente Financiero del Gobierno  
Federal)  
Certificados de la Tesorería de la  
Federación CETES  
Mayo, 2001

Waymire, Richard & Sawtell, Rick  
Aprendiendo Microsoft SQL Server  
7.0 en 21 días  
México, Prentice-Hall, 2000.

Yourdon, Edwor  
Yourdon Time Bomb 2000  
Printice Hall, 1998

## REFERENCIAS WEB

[agamenon.uniandes.edu.co/sistemas/6601.htm](http://agamenon.uniandes.edu.co/sistemas/6601.htm)

<http://www.generalhipotecaria.com>

<http://www.gestiopolis.com/recursos/experto/catsexp/pagans/fin/no%2013/creditoycobranza.htm>

<http://www.microsoft.com/méxico/default.asp>

<http://www.oracle.com>

<http://www.sybase.com>

[www.microsoft.com/sql](http://www.microsoft.com/sql)

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

[www.otri.us.es/recursosPHP/manual/odbc/importar\\_tablas.htm](http://www.otri.us.es/recursosPHP/manual/odbc/importar_tablas.htm)