



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**

---

---

**FACULTAD DE ESTUDIOS SUPERIORES  
ARAGON**

**“SISTEMA DE ADMINISTRACIÓN Y CONTROL  
DE CUENTAS DE AHORRO”**

**T E S I S**

**QUE PARA OBTENER EL TÍTULO DE  
INGENIERO EN COMPUTACIÓN**

**PRESENTA:**

**CRUZ RONQUILLO GERMAN**

**Director de Tesis:**

**M.I. Juan Carlos Roa Beiza**



**MÉXICO, SAN JUAN DE ARAGON, EDOMEX 2011**



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.



## ÍNDICE

---

---



## INDICE

### Capítulo 1

#### ENTORNO DEL PROBLEMA

1.1 Introducción .....	2
1.2 Conceptos básicos de operaciones financieras y sus medidas estadísticas ...	7
1.3 Proceso actual del registro de la información .....	10
1.4 Proceso actual del manejo y organización de la información .....	16
1.4.1 Almacenamiento Digital .....	17
1.5 Propuesta del nuevo sistema .....	20
1.5.1 Infraestructura (Hardware).....	20
1.5.2 Sistema de Software .....	21

### Capítulo 2

#### MARCO TEÓRICO

2.1 Características, ventajas y desventajas de las bases de datos relacionales....	24
2.1.1 Definición de Bases de Datos .....	24
2.1.2 Bases de Datos Relacional .....	24
2.1.3 Características de las Bases de Datos Relacionales .....	25
2.1.4 Cardinalidad de las Relaciones .....	26
2.1.5 Algebra Relacional .....	27
2.1.6 Cálculo Relacional .....	30
2.1.7 Normalización .....	31
2.1.8 Ventajas de las Bases de Datos Relacionales .....	34
2.1.9 Desventajas de las Bases de Datos Relacionales .....	35
2.2 Características, ventajas y desventajas de Desarrollo del sistema en el método de ciclo de vida clásico .....	36
2.2.1 Análisis del Sistema .....	37
2.2.2 Determinación de los requerimientos del sistema .....	38
2.2.3 Diseño del Sistema .....	42
2.2.4 Instrumentación .....	43
2.2.5 Pruebas .....	44
2.2.6 Implantación .....	44
2.2.7 Mantenimiento .....	44
2.2.8 Ventajas y desventajas del Método de ciclo de vida clásico .....	44
2.3 Características, ventajas y desventajas de Microsoft Access 2007 .....	47
2.3.1 Ventajas de Microsoft Access .....	47



2.3.2 Desventajas de Microsoft Access .....	50
2.4 Características, ventajas y desventajas de Visual Basic 6 .....	51
2.4.1 Origen de Visual Basic .....	51
2.4.2 Características Generales .....	52
2.4.3 Ventajas y desventajas de Visual Basic .....	54
2.5 Características, ventajas y desventajas de la Arquitectura Cliente-Servidor ..	57
2.5.1 Características del Modelo Cliente-Servidor .....	57
2.5.2 Clasificación por capas de la arquitectura Cliente-Servidor .....	58
2.5.3 Elementos de red dentro de la arquitectura Cliente-Servidor .....	62

### Capítulo 3

#### ANALISIS Y PLANTEAMIENTO DEL PROBLEMA

3.1 Identificación del Problema .....	68
3.2 Recopilación y Análisis de la Información .....	73
3.2.1 Alta de cliente nueva .....	73
3.2.2 Apertura de una nueva cuenta .....	75
3.2.3 Cancelación de una cuenta .....	76
3.2.4 Deposito en ventanilla .....	78
3.2.5 Retiro en ventanilla.....	79
3.2.6 Reporte mensual por movimientos de cuenta .....	80
3.2.7 Corte de caja diario .....	81
3.2.8 Reporte variable de saldos en caja por sucursal .....	81
3.3 Levantamiento de los requerimientos generales y particulares que debe cubrir la solución de software a construir .....	82
3.3.1 Requerimientos generales .....	82
3.3.2 Requerimientos Particulares .....	83
3.4 Identificar los posibles módulos de la aplicación .....	87
3.4.1 Definición de módulo .....	87
3.4.2 Características de un módulo .....	88
3.4.3 Identificación de módulos .....	89
3.5 Justificación de la metodología y del software a utilizar .....	96
3.5.1 Justificación de la metodología a utilizar .....	96
3.5.2 Justificación del software a utilizar .....	98

### Capítulo 4

#### DISEÑO Y CONSTRUCCION DE LA APLICACIÓN

4.1 Arquitectura de la aplicación .....	102
4.1.1 Pantalla de inicio .....	102

---

---



4.1.2 Login	103
4.1.3 Registros	103
4.1.4 Operaciones	105
4.1.5 Listados	105
4.1.6 Datos Estadísticos	105
4.1.7 Consulta Avanzada	105
4.1.8 Utilidades	106
4.1.9 Ayuda	106
4.1.10 Salir	106
4.2 Diagramación	107
4.2.1 Diagrama de Contexto	107
4.2.2 Diagramas de Proceso	109
4.2.3 Módulo de Administración	112
4.2.4 Módulo de Reportes	114
4.2.5 Módulo de Búsqueda	115
4.2.6 Módulo de Mantenimiento	116
4.2.7 Modelado de una Base de Datos	117
4.2.8 Diagramas de Flujo del Sistema de Administración y Control de Cuentas de Ahorro	125
4.2.9 Diccionario de Datos	150
4.3 Diseño de la Base de Datos	157
4.4 Desarrollo de la interfaz de usuario	171
4.5 Integración, pruebas y mantenimiento	181
4.6 Generación de reportes	189
4.6.1 Reporte de listado de Clientes	190
4.6.2 Reporte de listado de Empleados	191
4.6.3 Reporte de listado de Cuentas	191
4.6.4 Reporte de Datos Estadísticos de Movimientos	192
<b>CONCLUSIONES</b>	<b>199</b>
<b>BIBLIOGRAFÍA</b>	<b>203</b>

---

---

La presente tesis está dedicada a esa persona tan especial que conocí y cambió mi vida, gracias por cambiar mi mundo y pintar de colores aquello que estaba gris.

# **A G R A D E C I M I E N T O S**

**Quiero agradecer primero a JesúsCristo, por permitirme seguir en el camino y prestarme vida para continuar con mis sueños.**

**Agradezco a la vida, por ponerme esos retos y enseñarme la verdadera esencia del vivir, gracias de los buenos y excelentes momentos que me permitieron conocerme y evolucionar como ser humano, por dejarme aprender y también por dejar que otros aprendan sobre mi persona.**

**Agradezco a mis padres, por el encauce, la dedicación y la esperanza que han puesto sobre su hijo, gracias por la confianza, el cariño, el amor y el apoyo brindado durante todo este tiempo, espero que para ustedes sea motivo de orgullo y que el éxito que he tenido, sea también éxito de ustedes y les llene de felicidad este momento.**

**A mis hermanos, por todo el apoyo brindado en toda mi vida, porque estamos juntos y porque hemos sabido sortear los contratiempos surgidos a través de nuestras vidas, sabiendo que siempre estamos el uno o el otro, para ayudar a enfrentar los obstáculos.**

**Gracias a la UNAM, por darme cobijo durante toda mi carrera, por permitirme aprender, y por la excelente oportunidad que me brindó al permitirme conocer los diferentes ángulos del ambiente universitario.**

**A mis amigos y compañeros, por su compañía, sus consejos y el apoyo que me han brindado durante las diferentes etapas de mi vida, por lo buenos momentos que pasamos y por todas las experiencias que hemos tenido, gracias.**

**A mis profesores por sus enseñanzas durante toda la carrera, por el apoyo extendido y a los conocimientos compartidos para mi forjamiento profesional.**

**... A todos Gracias.**





# **CAPÍTULO 1**

## **ENTORNO DEL PROBLEMA**



## 1.1 INTRODUCCIÓN.

A manera de preámbulo de la presente Tesis, resulta necesario en primera instancia conocer la actividad principal de la Sociedad de Ahorro; las necesidades que éste a su vez debe satisfacer a sus clientes para poderles brindar un excelente servicio, así como los productos que dicha Sociedad de Ahorro ofrece a éstos.

Es de suma importancia tener en cuenta, que la Sociedad de Ahorro brinda servicios de administración y control de cuentas de ahorro a diferentes clientes; entendiéndose como cuenta de ahorro al depósito ordinario a la vista (producto pasivo), en la que los fondos depositados por el cuentahabiente tienen disponibilidad inmediata.

La Sociedad, se ha encargado del buen manejo y control de las cuentas de ahorro, procurando que los clientes dispongan en cualquier momento de su capital invertido sin tener algún contratiempo. Para dicha institución el objetivo primordial es otorgar el mejor servicio y administrar de manera responsable los recursos financieros de sus clientes.

Sin embargo, la institución reporta pérdidas anuales ya que la administración y control de las diferentes cuentas de sus clientes, son realizadas manualmente. Para dicho proceso es necesario invertir una gran cantidad de horas hombre y recursos consumibles (hojas, lápices, bolígrafos, etc.).

Como es sabido, el manejar la documentación de forma manual genera pérdidas de información, debido a fallas humanas provocando más errores al momento de procesar

---

---



los datos. Por lo tanto, existen muchos retardos en la disponibilidad de ésta, generando pérdida de tiempo y dinero para la Sociedad de Ahorro y sus clientes.

Es de vital importancia para la Sociedad de Ahorro el evitar dichas pérdidas anuales, ya que se maneja una gran cantidad de clientes, y un considerable capital invertido por éstos.

La Sociedad de Ahorro lleva a cabo la tarea de manera básica y rudimentaria, aunque en el mercado existen programas computacionales dedicados al control de las finanzas y administración. Dichos programas son difíciles de implantar por que es necesario adquirir equipo sofisticado y costoso, además se requiere la capacitación de empleados, el valor del software es elevado y en ocasiones no cubre los requerimientos necesarios, ya sea porque los programas son muy complejos o no contienen los rubros financieros a evaluar.

En virtud de lo anterior y debido al aumento de clientes captados por la Sociedad de Ahorro, la presente Tesis tiene como propósito fundamental el desarrollo de un sistema automatizado, eficiente y apropiado para cubrir las necesidades requeridas, llevando a cabo el proceso de administración y control de las diversas cuentas de ahorro de sus clientes.

El objetivo principal del sistema es mejorar el registro, el proceso y la búsqueda de la información, así como de mantenerla disponible en el momento que se necesite; este sistema permitirá controlar y manejar las cuentas de ahorro, reduciendo tiempos y costos.

---

---



El sistema contendrá la información de los movimientos realizados por los clientes, asimismo, buscará que se lleve un control automático y riguroso de estos, así como sus diversas cuentas. También incluirá una base de datos de los empleados.

Cabe mencionar que dicha base de datos también permitirá a la Sociedad de Ahorro contar con una herramienta que le facilite la incorporación de nuevos cuentahabientes de forma rápida y organizada, reduciéndose de esta manera los errores humanos.

El sistema permitirá el acceso a la información almacenada para que la Sociedad de Ahorro, mediante técnicas estadísticas, planee a futuro las políticas y acciones que convengan a ésta.

La información de los clientes se encontrará organizada en la base de datos, reduciéndose de esta manera los riesgos de pérdida de información (uso indebido de datos personales de los clientes, así como el acceso a información de los montos de las cuentas de éstos), al capturarse de forma directa.

El sistema, reducirá los costos de la administración de las cuentas de ahorro (número de horas hombre y gastos por insumos), incrementará la productividad de la institución (permitiendo a su personal atender un mayor número de clientes), facilitará la generación de reportes mensuales y el manejo de la información contenida en la base de datos, haciendo uso de las herramientas como hoja de cálculo y procesador de textos.

El método que se empleará para el desarrollo del sistema será el de Ciclo de Vida Clásico, ya que cumple con todos los puntos necesarios para el desarrollo del sistema.

---

---



Se llevara a cabo una investigación preliminar, ya que el cliente presenta un problema y busca resolverlo.

Se requiere revisar la petición del cliente, el estudio de factibilidad (técnica, económica y operacional) y la aprobación de la petición.

El objetivo del sistema es resolver la problemática del cliente de la manera más eficiente, optimizando los recursos del mismo.

Se realizarán las pruebas correspondientes al sistema, para posteriormente implementarlo. El cliente evaluará el mismo e informará si cumple con la resolución de su problema.

La presente Tesis se realizará cubriendo todos los requisitos académicos que se solicitan en la Universidad Nacional Autónoma de México, y constara de cuatro capítulos.

En el capítulo primero se abordara el entorno del problema, los conceptos básicos de operaciones financieras, el proceso actual de registro de la información, el actual manejo y organización de ésta. Así mismo incluirá los costos de operación y propuesta del nuevo sistema.



En el capítulo segundo se revisara el marco teórico, así como las características, ventajas y desventajas de las bases de datos relacionales, el desarrollo del sistema en el método de ciclo de vida clásico, las características de Microsoft Access 2007, de Visual Basic 6 y la Arquitectura Cliente-Servidor.

En el capítulo tercero se identificará el problema, se realizará el análisis y planteamiento del mismo, se hará la recopilación y el análisis de la información, el levantamiento de los requerimientos generales y particulares que debe cubrir la solución tecnológica de software a construir. También se identificarán los posibles módulos de la aplicación y la justificación de la metodología y del software a utilizar.

Por último en el capítulo cuatro se llevara a cabo el diseño y la construcción de la aplicación (su arquitectura y diagramación), se diseñará la Base de Datos, el desarrollo de la interfaz de usuario, así como la integración, pruebas y mantenimiento de la misma; para finalmente generar los reportes que se requieran.



## 1.2 CONCEPTOS BÁSICOS DE OPERACIONES FINANCIERAS.

Las operaciones financieras ejecutadas en un sistema de ahorro deben estar claramente definidas y delimitadas, con el fin de evitar malas interpretaciones, y resolver cualquier aclaración futura dentro del desarrollo del sistema, así como en su posterior entrega, uso y mantenimiento.

Un sistema de cuentas de ahorro puede formar parte de un sistema bancario o de algún otro esquema, así como también existir como una entidad independiente. De cualquier forma, los conceptos son los mismos.

Una operación bancaria es un movimiento que afecta directamente a una cuenta bancaria con un cambio en su saldo, ya sea una disminución o un incremento. Cada operación conlleva un número de la misma que es único e irrepetible. De igual forma, con la realización de cada operación se deben emitir dos registros iniciales, uno es el denominado comprobante del cliente y otro es el registro interno de la operación en los libros del banco.

El comprobante del cliente es un documento el cual ampara la operación realizada, debe mostrar todos los datos pertinentes a dicha operación según sea el caso, como cuenta de origen, cuenta de destino, monto, número de operación, nombre del cliente, nombre y número del cajero, etc. Este comprobante sirve al cliente como amparo de cada operación realizada por el, deberá contar con medidas de autenticidad, como sello bancario y firma del cajero, y es responsabilidad del cliente conservarlo. Cada comprobante de cliente emitido usualmente conlleva una copia, la cual es almacenada



por el banco como registro secundario de dicha operación, ya que el registro creado por el cajero directamente en el sistema se considera la fuente primaria.

Una cuenta de ahorro es un registro único, bajo el cual un cliente puede almacenar capital, y de igual forma retirarlo. Un cliente puede contar con varias cuentas de ahorro, pero una cuenta no puede estar asociada a más de un cliente. Cada operación bancaria se asocia a una cuenta, que es el sujeto de dicha operación.

El saldo de una cuenta de ahorro es una medida absoluta del balance entre los movimientos de depósito y retiro que un cliente ha realizado. Por ser productos exclusivos de ahorro, el saldo de una cuenta nunca deberá ser inferior a cero, ya que esto constituiría un producto crediticio.

Un depósito en efectivo a cuenta es una operación bancaria mediante la cual el cliente entrega una cantidad de dinero al banco para su almacenamiento, referenciando a una cuenta de ahorro. Dicha cantidad se suma al saldo previo de la cuenta en forma inmediata.

Un retiro es la operación mediante la cual el cliente recupera dinero en efectivo de una cuenta propia.

Para efectuar dicha operación, se debe realizar una consulta de saldo previa, y verificar que la cantidad solicitada en el retiro no sea excedente del saldo de la cuenta.





El cobro de una comisión es una operación mediante la cual el banco provoca una disminución en el saldo del cliente, adjudicando dicha cantidad como de su propiedad, por concepto de servicios prestados o como penalización por incumplimiento del cliente de una o varias de las normas establecidas en el contrato.

Existen dos actores principales dentro de un sistema de cuentas de ahorro, el cliente y el banco. El cliente se define como una entidad con necesidad de almacenar una cantidad variable de capital, en al menos una ocasión y con intenciones de disponer del total o parte de su dinero almacenado en algún momento. El cliente debe seguir los procedimientos establecidos con el banco para la realización de las operaciones, así como llenar la documentación requerida para las mismas.

El banco es una institución la cual se encargara de recibir y almacenar el dinero de varios clientes, es responsable de llevar la información al corriente del saldo de cada una de las cuentas, así como de establecer los procedimientos necesarios para realizar todas las posibles operaciones financieras sobre las cuentas. De igual manera, el banco es el responsable de la presentación de reportes cronológicos acerca del historial de movimientos del cliente.

Dentro del banco, existen a su vez actores menores encargados del manejo de las operaciones en ventanilla. Dichos agentes son conocidos como cajeros, y son los responsables de efectuar el registro inicial de las operaciones solicitadas por los clientes, emitir los comprobantes necesarios y ejecutar dichas operaciones.



### **1.3 PROCESO ACTUAL DEL REGISTRO DE LA INFORMACIÓN.**

En cualquier proceso la información con la que se cuenta tiene que ser útil, fácil de encontrar y estar disponible para toda persona que la requiera, esto dependerá en gran medida de la forma cómo se le maneja y organiza.

La información que se maneja en la Sociedad de Ahorro tiene que ser registrada de forma inmediata, anotando todos los datos correctamente para garantizar el buen manejo de las cuentas, se requiere que la información se encuentre disponible cuando se necesita y por ende lograr la satisfacción del cliente. Por lo anterior, se requiere tener un control estricto de los datos de los clientes y de sus cuentas.

La Sociedad de Ahorro brinda a sus clientes principalmente cuentas de ahorro a plazos y sistemas de monedero electrónico. Las cuentas se manejan en pesos, dólares o euros y los movimientos en las cuentas pueden ser de depósito o retiro.

Actualmente el registro de toda la información que se maneja en la Sociedad de Ahorro se lleva a cabo manualmente, específicamente en hojas de cálculo. Sin embargo, debido al crecimiento de la empresa, la administración se ha vuelto cada vez más complicada al tener una gran cantidad de información y no poder actualizar todos los registros cuando se necesita.



Derivado de lo anterior, se han generado pérdidas de información y dinero, y también se ha puesto en duda la credibilidad de la Institución, por lo que se hace urgente una mejor forma de agilizar y organizar la información.

Para analizar el proceso actual de registro de la información es necesario mencionar las diferentes áreas de la Sociedad de Ahorro que se involucran en el manejo de la información y que son las siguientes: Véase Figura 1.3.1.

**Dirección General:** Se encarga de formular planes y estrategias para lograr los objetivos de la Institución.

**Administración y Finanzas:** Se encarga de la administración general de los clientes y los movimientos de sus cuentas.

**Sistemas Informáticos:** Se encarga de satisfacer las necesidades informáticas de la empresa y ofrecer soluciones en el manejo de la información.

**Recursos Humanos:** Lleva el registro de todos los aspectos relacionados con el personal de la empresa.



**Figura 1.3.1 - Áreas de la Sociedad de Ahorro.**

El registro de los clientes asociados a la Sociedad de Ahorro en el año en curso se registra en una hoja de cálculo por mes, en la que se incluye: el número de cliente, nombre del cliente, RFC, dirección, teléfono, sexo, delegación, antigüedad en la Institución y situación actual (si se encuentra activo o es dado de baja). Los clientes que son dados de baja en el mes en curso se subrayan de un color distinto para diferenciarlos y asimismo se anota la fecha en la que dejan de pertenecer a la Sociedad de Ahorro para no ser incluidos en el siguiente mes. Véase Figura 1.3.2.



## SISTEMA DE ADMINISTRACIÓN Y CONTROL DE CUENTAS DE AHORRO



NUM. CLIENTE	NOMBRE CLIENTE	RFC	DIRECCIÓN	TELEFONO	SEXO	DELEGACIÓN	ANTIGÜEDAD	SITUACIÓN
COO1-2011	KEVIN SERVÍN GONZÁLEZ	SE GK741212	AV. TAMAULIPAS No. 1240 colonia CORPUS CHRISTY	54-82-40-62	M	ALVARO OBREGON	02/05/2009	ACTIVO
COO2-2011	VÍCTOR ALEXIS GONZÁLEZ	GOSV782101	PROLONGACION SN. ANTONIO No. 705 colonia LOMAS DE BECERRA	56-62-64-94	F	COYOACÁN	03/05/2010	ACTIVO
COO3-2011	ELIA MUÑOZ LÓPEZ	MULE510825	RIO MIXCOAC No.343 colonia AXOTLA	56-62-24-81	F	TLALPAN	04/04/2010	ACTIVO
COO4-2011	DIANA GOMEZ PEREZ	GOPD780504	INSURGENTES SUR No. 2358 colonia CHIMALISTAC	55-70-60-02	F	TLAHUAC	05/04/2008	BAJA 01/05/2012
COO5-2011	ALEXIS SANTOS DÍAS	SADA650903	AV. VASCO DE QUIROGA No. 1552 colonia SANTA FE	56-35-49-91	M	CUAUHTÉMOC	06/06/2010	ACTIVO
COO6-2011	JUAN GONZALEZ GALVÁN	GOGJ510305	CALZADA LAS AGUILAS No. 969 colonia AMP. LAS AGUILAS	55-70-86-28	M	TLALPAN	07/07/2010	ACTIVO
COO7-2011	IRMA DUARTE MORALES	DUMI690403	AV. VASCO DE QUIROGA No. 1845 colonia SANTA FE	56-60-22-58	F	IZTAPALAPA	08/04/2010	ACTIVO
COO8-2011	CARMELA GONZALEZ MORAN	GOMC761207	AV. SANTA LUCIA No. 814 colonia HOGAR Y REDENCION	56-60-29-06; 56-60-29-13	M	COYOACÁN	15/01/2004	ACTIVO
COO9-2011	RAFAEL JUAN MUÑOZ G	MULR820907	AV. 5 DE MAYO No. 467 (ANTES MILPA ALTA No.11) colonia MERCED GOMEZ cp 01600	55-95-53-63	F	CUAUHTÉMOC	30/05/2007	ACTIVO
CO10-2011	VICTOR SERVÍN VITE	SEVV77020	AVENIDA TOLUCA No. 147 colonia OLIVAR DE LOS PADRES cp 01780	55-59-48-23	M	BENITO JUÁREZ	15/08/2008	ACTIVO

**Figura 1.3.2 - Registro de Clientes**

Como se muestra, el registro de la información está hecho en forma desordenada y no se tiene un control de las personas que son dadas de baja ya que si se desea saber el número de clientes que han sido dados de baja en el año en curso se tendrían que buscar mes con mes y sería un proceso tardado.

Es necesario llevar un control estricto de los empleados de la Sociedad de Ahorro, ya que cada empleado, dependiendo de su cargo y las funciones que se le asignen, requiere tener ciertos privilegios para control y acceso de la información. Un empleado perteneciente a la Dirección General gozará de todos los privilegios y claves de acceso para el manejo de las cuentas. Los empleados pertenecientes al área de Administración y Finanzas deberán registrar su número de empleado como responsables de los movimientos que realicen con las cuentas. Sin embargo, actualmente no existe un



control de las cuentas de acceso ya que cualquier empleado que así lo desee, puede ingresar a los registros y modificar los datos.

El registro de los empleados es responsabilidad del área de Recursos Humanos. Actualmente se registran los datos de los empleados de toda la Sociedad de Ahorro en una hoja de cálculo en la que se incluye: código de empleado, nombre del empleado, RFC, dirección, teléfono, sexo, delegación, antigüedad en la Institución y cargo. Asimismo, con la finalidad de llevar un control del número de empleados que pertenecen a las diferentes áreas se realiza el registro de empleados por área. Véase Figura 1.3.3.

COD. EMPLEADO	NOMBRE EMPLEADO	RFC	DIRECCIÓN	TELEFONO	SEXO	DELEGACIÓN	ANTIGÜEDAD	CARGO
EMP001	JUAN GONZÁLEZ GALVÁN	GOGJ510305	AV. SANTA LUCIA No. 814 colonia HOGAR Y REDENCION	56-60-29-06	M	TLALPAN	01/06/2006	DIRECCIÓN
EMP002	MAGALI SERVIN CASAS	SECM850403	AV. 5 DE MAYO No. 467 Colonia MERCED GOMEZ	55-95-56-56	F	CUAUHTEMOC	30/03/2010	REGISTRO
EMP003	VICTOR SANCHEZ PEREZ	SAPV680304	AVENIDA TOLUCA No. 147 colonia OLIVAR DE LOS PADRES	55-95-53-64	M	COYOACAN	30/01/2011	CONTADOR
EMP004	ANGELA MARIA ESPINOZA	ESGA700402	CENTEOTL NÚMERO 267 colonia INDUSTRIAL SAN ANTONIO	55-96-43-65	F	CUAUHEMOS	15/11/2010	CONTADOR
EMP005	JUAN PABLO SANCHÉZ	SAN890504	AV. REFINERIA AZCAPOTZALCO, NÚMERO. 190 colonia SANTA INES	53-95-42-98	M	AZCAPOTZALCO	01/09/2010	REGISTRO
EMP006	MARIA MAGDALENA LOPEZ	LOMM790506	CERRADA DE AVE. DE LAS GRANJAS No. 61 colonia JARDIN AZPETAIA	55-45-53-33	F	TLAHUAC	30/07/2009	SISTEMAS
EMP007	ANDRÉS LAGOS FERNANDE	LAFAT90302	AV. DE LAS CULTURAS No. 33 ESQ. RENACIM. colonia UNIDAD HAB. EL ROSARIO	55-13-45-55	M	TLALPAN	15/03/2008	CONTADOR
EMP008	MARÍA GONZALEZ NUÑEZ	GONM761207	COLTONGO No. 67 colonia COLTONGO cp 02630	55-11-22-85	F	BENITO JUÁREZ	01/04/2007	SISTEMAS
EMP009	MARIA ANGELES VITE ZUÑIG	VIZM790403	CALLE OLIVOS No.54 COL. LA ESPERANZA	56-25-54-45	F	COYOACAN	15/06/2005	REGISTRO

Figura 1.3.3 - Registro de empleados.

El manejo de las cuentas de los clientes es responsabilidad del área de Administración y Finanzas. El registro de los movimientos cuentas de los clientes se guarda en un libro de hoja de cálculo por año y en éste último se guarda la información de los movimientos de las cuentas por mes incluyendo: número de cuenta, nombre del



cliente, tipo de cuenta, moneda, el monto inicial, cantidad, saldo final, fecha de la transacción, hora y número de empleado responsable de efectuar la transacción. Véase Figura 1.3.4.

	A	B	C	D	E	F	G	H	I	J	K
1	NUM.CTA.	NOMBRE CLIENTE	TIPO CUENTA	MONEDA	MONTO INICIAL	MOVIMIENTO	CANTIDAD	SALDO FINAL	FECHA	HORA	EMPLEADO
2	CTA001	KEVIN GONZALEZ C	AHORRO	PESOS	50000	RETIRO	2000	48000	04/05/2011	09:20	EMPO09
3	CTA002	ADRIANA CAMACHO G	PLAZOS	PESOS	30000	DEPOSITO	5000	35000	05/05/2011	10:35	EMPO10
4	CTA003	KAREN SANCHEZ F	AHORRO	DOLARES	450	RETIRO	100	350	12/05/2011	11:50	EMPO11
5	CTA004	DIANA GOMEZ PEREZ	AHORRO	PESOS	15000	RETIRO	2000	13000	07/05/2011	13:05	EMPO14
6	CTA005	ALEXIS SANTOS DÍAS	PLAZOS	PESOS	23456	DEPOSITO	3000	26456	04/05/2011	14:20	EMPO10
7	CTA006	CARLOS LAGOS OCHOA	MONEDERO	DOLARES	250	DEPOSITO	450	700	09/05/2011	15:35	EMPO14
8	CTA007	IRMA DUARTE MORALES	PLAZOS	EUROS	30000	RETIRO	1000	29000	10/05/2011	16:50	EMPO10
9	CTA008	ARACELI GARCIA ROSAS	PLAZOS	PESOS	850	DEPOSITO	3000	3850	06/05/2011	18:05	EMPO09
10	CTA009	JAVIER EESTRADA CASAS	MONEDERO	PESOS	15000	RETIRO	1000	14000	12/05/2011	09:20	EMPO17
11	CTA010	MAGALI SERVIN HERNAN	PLAZOS	PESOS	23456	RETIRO	456	23000	13/05/2011	10:35	EMPO14
12	CTA011	VICTOR CORONA OCHOA	MONEDERO	EUROS	3348	RETIRO	348	3000	09/05/2011	11:50	EMPO10
13	CTA012	ANGELA MARIA ESPINOZ	PLAZOS	PESOS	38723	DEPOSITO	3000	41723	15/05/2011	13:05	EMPO14
14	CTA013	JUAN PABLO SANCHEZ	MONEDERO	PESOS	46098	DEPOSITO	4000	50098	16/05/2011	11:20	EMPO17

Figura 1.3.4 - Movimientos de las Cuentas.

La forma actual de registro de la información es obsoleta, desordenada y si agregamos que se realiza por diferentes personas esto ocasiona:

- Pérdida de tiempo.
- Retraso en los reportes mensuales.
- Que los registros lleguen a duplicarse.
- Gastos innecesarios y pérdidas monetarias derivados de errores humanos.
- Molestia en los clientes.



## 1.4 PROCESO ACTUAL DEL MANEJO Y ORGANIZACIÓN DE LA INFORMACIÓN.

A raíz del crecimiento de la Sociedad de Ahorro, creció el número de clientes y en paralelo se acrecentó el volumen de la información. Resolver en un principio el almacenaje de todas las cuentas de los clientes fue tarea fácil, se decidió colocar toda la papelería en carpetas plásticas siguiendo la siguiente metodología.

- La información sería almacenada por año.
- El orden sería alfabéticamente.
- Las carpetas serían de distinto color de acuerdo al producto solicitado.

Esto resolvió el problema en su momento, pero con el incremento de los clientes, creció el número de carpetas y se optó por el siguiente proceso:

- Todas las carpetas se colocarían en anaqueles.
- Las carpetas más antiguas se irían a bodega.
- En bodega estarán almacenadas las carpetas en cajas de cartón de acuerdo al producto y al año.

Con esta metodología se trabajó un tiempo, pero llegó el momento en que la bodega fue insuficiente para almacenar la información, el crecimiento de papelería, carpetas y cajas de cartón, trajo consigo rentar otro espacio físico para poder alojar toda esa cantidad de información. Otra problemática que surgió, fue que cuando se tenía que

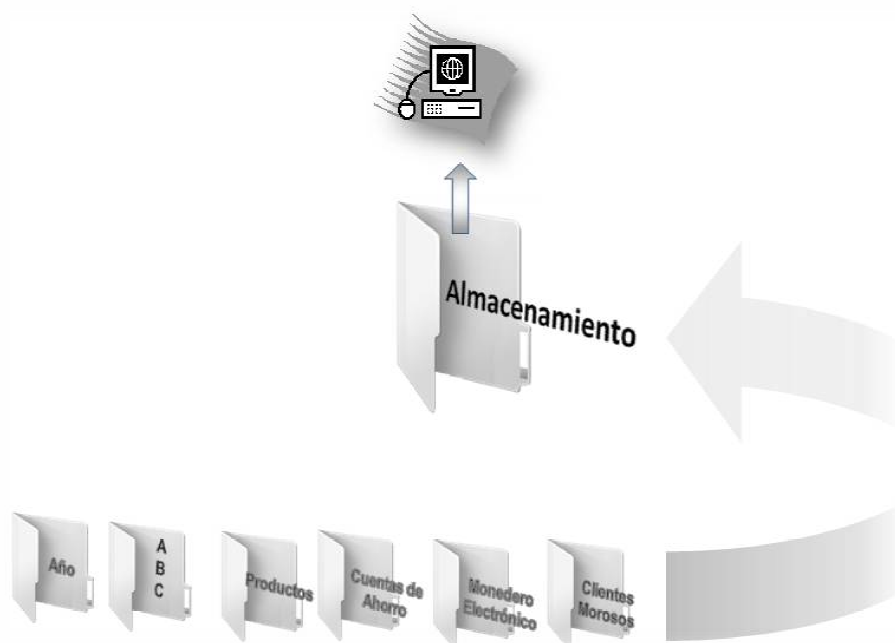


resolver alguna circunstancia, había que revisar todas las cajas y poder rastrear la información y localizar donde se genero el error, esto demoraba bastante tiempo, pues había que desmembrar todos los paquetes almacenados anteriormente y existía gran pérdida de información. Resultando pérdidas monetarias para la empresa.

### 1.4.1 Almacenamiento Digital

La Sociedad de Ahorro debería resolver esta problemática cuanto antes, ya que a ese ritmo la empresa se metería en dificultades en un corto plazo.

En la figura 1.4.1 se muestra la forma de almacenar la información en computadoras, en carpetas digitales, toda la información sería guardada en los equipos del personal.

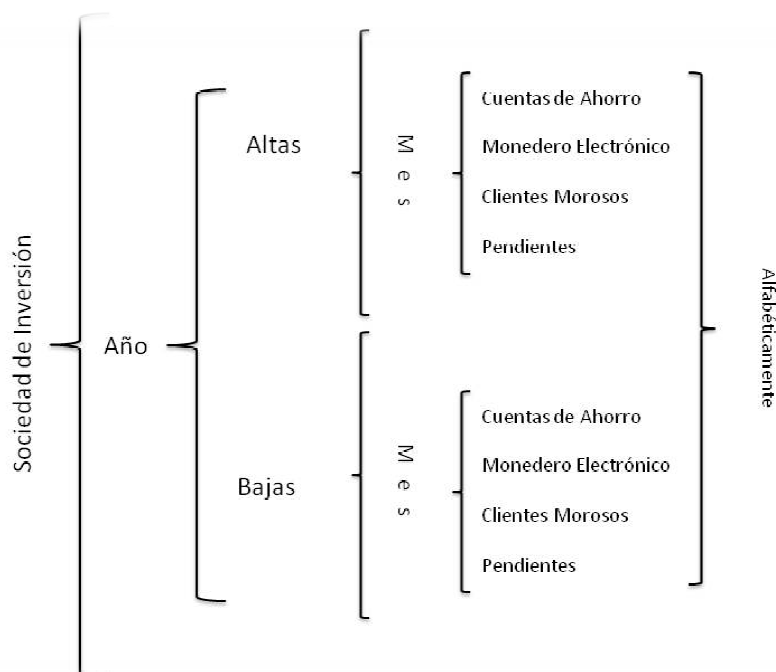


**Figura 1.4.1 Esquema de almacenamiento basado en carpetas digitales.**



De esta forma se logró reducir en un 85% el espacio en almacenamiento de la información y el manejo de la misma fue más accesible. La organización de las carpetas se estandarizó para que todo el personal conociera como buscar algún dato y en dado caso de que la persona se ausentara, otro colaborador pudiera tener acceso rápido al equipo. A la par con esta nueva metodología, se capacitó al personal para que tuviera los conocimientos para poder manejar el equipo, la capacitación consistió en creación de archivos, carpetas y respaldos, manejo de computadora y hoja de cálculo; mediante lo cual se logró reducir las horas-hombre y las horas-maquina.

La figura 1.4.2 muestra la organización de la información, esta se clasifica por año, alta o baja, mes, tipo de cuenta y finalmente por orden alfabético.

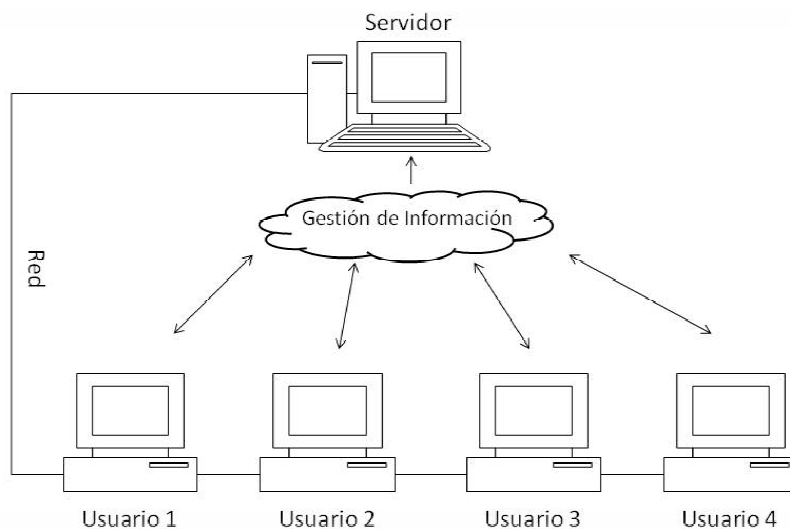


**Figura 1.4.2 Esquema de Organización de la información.**

En caso de realizar alguna aclaración, se solicita al cliente su fecha de registro, se localiza la carpeta del año en que fue generada la cuenta, el mes, el tipo de servicio y se localizaba el nombre del cliente.

La información se almacenaba en los discos duros de las computadoras del personal, de acuerdo a la organización anteriormente mencionada (figura 1.4.2). La acumulación de información comenzó a saturar los discos duros, presentado fallas tales como, congelamiento del sistema, lentitud en las operaciones y posteriormente incapacidad para almacenar más archivos.

Por lo tanto se decidió establecer una máquina como servidor general como se muestra en la figura 1.4.3, en este servidor se almacenan todos los archivos y estos pueden ser consultados por el personal de la empresa.



**Figura 1.4.3 - Computadora establecida como Servidor.**

Actualmente esta es la forma como se trabaja en la Sociedad de Ahorro, ejemplificado en la figura 1.4.3. Toda la información es almacenada en el equipo-servidor. Con esta forma de trabajo cualquier persona los puede acceder vía red, ya no existen claves de acceso, niveles de usuarios o protección a modificación de archivos, por lo que la información es vulnerable a cualquier error o mal uso.



## 1.5 PROPUESTA DEL NUEVO SISTEMA

La nueva propuesta basada en un sistema de administración y control de cuentas de ahorro, como cualquier producto de software requiere para su funcionamiento de computadoras y sus periféricos, y adicionalmente una infraestructura de red.

Aunque el objetivo de estudio es el sistema de software, es necesario abordar su infraestructura ya que esto permitirá a la institución dimensionar la erogación total de la implementación de las nuevas tecnologías.

### 1.5.1 Infraestructura (Hardware)

La nueva propuesta considera instalar una pc y una impresora de matriz (impresora necesaria para imprimir en papel carbón) por cajero –o según lo amerite una para dos cajeros-, en cada sucursal; así como dos adicionales para atención a usuarios –altas, actualizaciones- y una más para apoyo administrativo. Los equipos operarán con el sistema operativo Microsoft Windows Vista, que es el normalmente utilizado en otras actividades. Los equipos operarán con procesadores Intel de dos núcleos, de 1.9 GHz. RAM de 2 GB, disco duro de 500 GB, tarjeta de red con velocidad de 100Gb. Los equipos de conectividad en red deberán ser concentradores de 8 puertos de 100Gb c/u. Se recomienda un reacondicionamiento de las líneas eléctricas a fin de proteger los equipos con variaciones de voltaje. De igual forma se recomienda que se implemente el servicio de enlaces dedicados a fin de mantener comunicación para intercambio de información entre las sucursales y la oficina matriz. Será importante para el envío cotidiano de la información de movimientos así como para reducir gastos de telefónica usual por telefonía IP.



En el área de contaduría deberán instalarse pc`s e impresoras laser, conectadas en red, dedicadas para la aplicación del sistema de cuentas de ahorro. La institución ya cuenta con un servidor para el manejo de información de otras actividades de la institución, por lo que solo requería de la ampliación de espacio en disco a fin de que pueda almacenar la información del nuevo sistema.

### **1.5.2 Sistema de software.**

Como parte de los requerimientos del sistema, y ya que entre los de cómputo existentes se cuentan con los productos de Microsoft –Windows, Office- y sus respectivas licencias empresariales, la base de datos se desarrollará en Microsoft Access debido a que el soporte y mantenimiento son más sencillos y no se tiene que tener personal especializado, ya que esto incrementaría el sueldo de un personal certificado y el Front-End se implementará en VB 6.0, del cual la empresa cuenta también con licencia.

A fin de que el sistema se ajuste a las necesidades del proceso administración y control de cuentas de ahorro, deberá operar las siguientes actividades:

- Alta de cuentahabientes. –Incluir datos personales (Nombre, Sexo, Dirección).
- Alta de Empleados: cajeros, gerentes, contadores/analistas, Incluir datos personales a fin de que el sistema pueda ser manipulado por el personal de la Sociedad de Ahorro en oficinas centrales y sucursales existentes.
- Alta de Cuentas: Ahorro normal, ahorro fácil, a plazos y que podrán hacerse en moneda nacional o extranjera (Dólares o Euros). El cuentahabiente podrá tener una o varias cuentas en los tipos y monedas que ofrece la sociedad de ahorro. El sistema automáticamente relacionará al usuario con las cuentas y estas con los empleados por medio de cuentas, siendo estas únicas, una por cada cliente. Captura de movimientos. El sistema permitirá a los cajeros registrar los



movimientos hechos por los cuentahabientes, durante las horas y días establecidos por la sociedad de ahorro. Los movimientos establecidos para todas los tipos de cuentas son: Deposito y retiro, en las tipos de monedas señalados arriba. De las operaciones efectuadas el sistema imprimirá recibos con los siguientes datos: Nombre del cuentahabiente, número de cuenta del cuentahabiente, tipo de cuenta, tipo de movimiento, fecha de la operación, monto de la operación.

- El sistema permitirá la consulta de información de los usuarios, cuentas y empleados dedicados a las actividades propias del sistema, tanto para efectos de atención de los cuentahabientes como para uso de la institución.
- La baja de cuentas, clientes y empleados de acuerdo a los criterios establecidos por la sociedad de ahorro.
- Generación de reportes impresos para informar a los cuentahabientes del estado que guardan sus cuentas de ahorro, así como reportes impresos o en formato .xls para facilitar el manejo de la información solicitada en la toma de decisiones (análisis estadísticos, tendencias de mercado, etc.).
- Actualización de los montos de las cuentas en función de los movimientos hechos, las tasas de interés establecidas en el mercado actual, así como penalizaciones por no cumplir con el monto mínimo de ahorro.
- Utilerías que faciliten a los operadores el manejo de la información contenida en la base de datos mediante el acceso a utilerías como Notepad, calculadora, Excel y Word.
- El sistema asegurará la integridad y seguridad de la información mediante el diseño un esquema relacional de la base de datos adecuado, implementación de elementos de seguridad, privilegiando los puestos y la implementación de elementos de recuperación, reparación y de respaldo de la base de datos, con las herramientas ofrecidas por Microsoft Access.



## **CAPÍTULO DOS MARCO TEORICO**



## **2.1 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE LAS BASES DE DATOS RELACIONALES.**

### **2.1.1 Definición de base de datos.**

La base de datos puede definirse como una colección de datos interrelacionados almacenados en conjunto, sin redundancias perjudiciales o innecesarias; su finalidad es de servir a una aplicación o más, de la mejor manera posible; los datos se almacenan de modo que resulten independientes de los programas que los usan; se emplean métodos bien determinados para incluir datos nuevos y para modificar ó extraer los datos almacenados .Dícese que un sistema comprende una colección de base de datos cuando estos son totalmente independientes desde el punto de vista estructural.

### **2.1.2 Base de datos relacional.**

Una base de datos relacional es aquella que cumple con el modelo relacional, el cual es el más utilizado en la actualidad para implementar bases de datos ya planificadas. Permite establecer interconexiones (relaciones) entre los datos (que están guardados en tablas) y a través de dichas conexiones relacionar los datos de las tablas, de ahí proviene su nombre: "Modelo Relacional".

Los sistemas relacionales son importantes porque ofrecen ventajas tales como: simplicidad, generalidad y las consultas de información se especifican de forma sencilla. Las tablas son un medio para organizar la información de una forma más compacta y al mismo tiempo acceder a ella desde una interfaz.





### 2.1.3 Características de las bases de datos relacionales

- Las bases de datos relacionales están constituidas por una o más tablas que contienen la información ordenada, organizada y que además cumplen con las siguientes características básicas:
  - Una tabla sólo contiene un ID único e irrepetible.
  - El nombre de los campos de una tabla es distinto.
  - Cada registro de la tabla es único.
  - El orden de los registros no está determinado.
  - Para cada campo existe un conjunto de valores posible.
- La planificación de la estructura de la base de datos, en particular de las tablas es vital para la administración. Cada tabla tiene una llave primaria, un identificador único, compuesto por una o más columnas. La mayoría de las llaves primarias están formadas por una única columna (atributo), es decir, la llave primaria es un conjunto de uno o más atributos que sirve para identificar de forma unívoca a una tupla.

Para establecer la relación entre dos tablas es necesario incluir, en forma de columna, en una de ellas la llave primaria de la otra. A esta columna se le llama llave secundaria. La llave secundaria nos permite expresar relaciones entre objetos.

- Integridad Referencial es otra característica importante. Se entiende por integridad referencial a las reglas que se establecen para mantener las relaciones entre las tablas cuando se agregan, cambian o eliminan registros.

Al exigir la integridad referencial, se impide a los usuarios que agreguen registros a la tabla relacionada para la cual no hay llave principal, cambiar los valores de la tabla



principal que darían lugar a registros huérfanos en la tabla relacionada, así como eliminar registros de una tabla principal cuando hay registros relacionados coincidentes.

Los términos formales del modelo relacional a menudo son sustituidos por otros de uso común, debido a que estos términos son demasiado abstractos para ser usados en la práctica. La relación se muestra en la tabla 2.1.3.1.

<b>Término relacional formal</b>	<b>Equivalente informal</b>
Relación	Tabla
Tupla	Fila o Registro
Cardinalidad	Número de filas o registros
Atributo	Columna o campo
Grado	Número de columnas o campos
Llave Primaria	Identificador único
Dominio	Conjunto de valores permitidos para el atributo.

**Tabla 2.1.3.1 - Términos formales e informales del modelo relacional.**

#### **2.1.4 Cardinalidad de las relaciones**

El tipo de cardinalidad se representa mediante una etiqueta en el exterior de la relación, respectivamente: "1:1", "1:N" y "N:M", aunque la notación depende del lenguaje utilizado, la que más se usa actualmente es el unificado. Otra forma de expresar la



cardinalidad es situando un símbolo cerca de la línea que conecta una entidad con una relación:

- **"0"** si cada instancia de la entidad no está obligada a participar en la relación.
- **"1"** si toda instancia de la entidad está obligada a participar en la relación y, además, solamente participa una vez.
- **"N", "M", ó "\*" "** si cada instancia de la entidad no está obligada a participar en la relación y puede hacerlo cualquier número de veces.

### 2.1.5 Álgebra relacional

El álgebra relacional es un conjunto de operaciones que describen paso a paso como computar una respuesta sobre las relaciones, tal y como éstas son definidas en el modelo relacional, denominada de tipo procedural.

Describe el aspecto de la manipulación de datos. Estas operaciones se usan como una representación intermedia de una consulta a una base de datos y debido a sus propiedades algebraicas, sirven para obtener una versión optimizada y eficiente de dicha consulta.

#### 2.1.5.1 Operaciones básicas

Cada operador del álgebra acepta una o dos relaciones y retorna una relación como resultado.  $\sigma$  y  $\Pi$  son operadores unarios, el resto de los operadores son binarios. Las operaciones básicas del álgebra relacional son:

- **Selección ( $\sigma$ )**



Permite seleccionar un subconjunto de tuplas de una relación (R), todas aquellas que cumplan la(s) condición(es) P, esto es:

$$\sigma_P(R)$$

- **Proyección ( $\Pi$ )**

Permite extraer columnas (atributos) de una relación, dando como resultado un *subconjunto vertical* de atributos de la relación, esto es:

$$PA1, A2, \dots, An(R)$$

donde,  $A1, A2, \dots, An$  son atributos de la relación R .

- **Producto cartesiano ( $\times$ )**

El producto cartesiano de *dos relaciones* se escribe como:

$$R \times S$$

y entrega una relación, cuyo *esquema* corresponde a una combinación de todas las tuplas de R con cada una de las tuplas de S, y sus atributos corresponden a los de R seguidos por los de S.

- **Unión ( $\cup$ )**

La operación

$$R \cup S$$

retorna el conjunto de tuplas que están en R, o en S, o en ambas. R y S deben ser *uniones compatibles*.

- **Diferencia ( $-$ )**

La diferencia de dos relaciones, R y S denotada por:

$$R - S$$

entrega todas aquellas tuplas que están en R, pero no en S. R y S deben ser *uniones compatibles*.



Estas operaciones son fundamentales en el sentido en que (1) todas las demás operaciones pueden ser expresadas como una combinación de éstas y (2) ninguna de estas operaciones pueden ser omitidas sin que con ello se pierda información.

### 2.1.5.2 Operaciones no básicas

- **Intersección ( $\cap$ )**

La intersección de dos relaciones se puede especificar en función de otros operadores básicos:

$$R \cap S = R - (R - S)$$

La intersección, como en Teoría de conjuntos, corresponde al conjunto de todas las tuplas que están en R y en S, siendo R y S *uniones compatibles*.

- **Unión natural ( $\bowtie$ ) (Natural Join)**

La operación unión natural en el álgebra relacional es la que permite reconstruir las tablas originales previas al proceso de normalización. Consiste en combinar las proyección, selección y producto cartesiano en una sola operación, donde la condición  $\theta$  es la igualdad Clave Primaria = Clave Externa (o Foranea), y la proyección elimina la columna duplicada (clave externa).

Expresada en las operaciones básicas, queda

$$R \bowtie S = \Pi_{A_1, A_2, \dots, A_n}(\sigma_{\theta}(R \times S))$$

Una reunión theta ( $\theta$ -Join) de dos relaciones es equivalente a:

$$R \bowtie_{\theta} S = \sigma_{\theta}(R \times S)$$

donde la condición  $\theta$  es libre.

Si la condición  $\theta$  es una igualdad se denomina EquiJoin.



- **División (/)**

Supongamos que tenemos dos relaciones  $A(x, y)$  y  $B(y)$  donde el dominio de  $y$  en  $A$  y  $B$ , es el mismo.

El operador división  $A / B$  retorna todos los distintos valores de  $x$  tales que para todo valor  $y$  en  $B$  existe una tupla  $(x, y)$  en  $A$ .

- **Agrupación (G)**

Permite agrupar conjuntos de valores en función de un campo determinado y hacer operaciones con otros campos.

### 2.1.6 Cálculo Relacional

El Cálculo relacional es un lenguaje de consulta que describe la respuesta deseada sobre una Base de Datos sin especificar cómo obtenerla, a diferencia del álgebra relacional que es de tipo procedural, el cálculo relacional es de tipo declarativo, pero siempre ambos métodos logran los mismos resultados.

Podemos distinguir, al menos, dos clases de cálculo relacional:

- **Cálculo relacional basado en tuplas. (TRC).**

Una consulta en TRC es de la forma:

$$\{T \mid \varphi(T)\}$$

donde  $T$  es una variable tipo tupla y  $\varphi(T)$  es una fórmula que describe a  $T$ . El resultado de esta consulta es todas las tuplas  $t$  para las cuales la fórmula es verdadera.



- **Cálculo relacional basado en dominios (DRC)**

Está constituido con los mismos operadores que el cálculo relacional de tuplas pero no hay tuplas sino variables dominio. Las expresiones del cálculo relacional de dominios son de la forma  $\{ (x, y, z, \dots) / P(x, y, z, \dots) \}$ . Donde  $x, y, z$  representan las variables de dominio,  $P$  representa una fórmula compuesta de átomos (igual que en el CRT). Los átomos del cálculo relacional de dominios tienen una de las siguientes formas:

- $(x, y, z) \in r$ , donde  $r$  es una relación con  $n$  atributos y  $x, y, z$  son variables de dominio o constantes.
- $x \theta y$ , donde  $x$  e  $y$  son variables de dominio y  $\theta$  es un operador de comparación aritmética ( $>$ ,  $<$ ,  $=$ ,  $\neq$ ). Es necesario que los atributos  $x$  e  $y$ , tengan dominios cuyos miembros puedan compararse mediante  $\theta$ .
- $x \theta c$ , donde  $x$  es una variable de dominio,  $\theta$  es un operador de comparación y  $c$  es una constante en el dominio del atributo  $x$ .

### 2.1.7 Normalización.

La normalización es un proceso que pretende conseguir tablas con una estructura óptima y eficaz. El proceso de normalización está basado en lograr la independencia de los datos respecto a las aplicaciones que los usan. Antes de empezar el proceso, se han de conocer las tablas que intervendrán y las relaciones que las unen. Si no se conocen a partir del análisis previo, se buscan todos los nombres (sustantivos) que han sido empleados en la definición del problema. Algunos de esos nombres serán las entidades, otros dependerán de ellas y serán los atributos. Otros no formarán parte ni de las entidades ni de los atributos, son parte del lenguaje necesario para describir el problema a solucionar mediante la creación de una base de datos. El proceso de



normalización se basa en la descomposición sin pérdida de las tablas que están en una forma normal inferior, obteniéndose una forma normal superior. El proceso de descomposición sin pérdida, significa que se ha de dividir o descomponer la tabla en otras con menor cantidad de atributos sin que haya pérdida de información.

La teoría de la normalización está basada en el concepto de formas normales. Se dice que una relación está en una forma normal particular si satisface cierto conjunto específico de restricciones.

#### **2.1.7.1 Formas Normales y dependencias funcionales.**

Las formas normales son aplicadas a las tablas de una base de datos. Decir que una base de datos está en la forma normal **N** es decir que todas sus tablas están en la forma normal **N**. En general, las primeras tres formas normales son suficientes para cubrir las necesidades de la mayoría de las bases de datos.

- Primera Forma Normal o 1FN:

La Primera Forma Normal, o 1FN, es la más elemental de todas. Una tabla está en 1FN si el valor que contiene un atributo de un registro, un campo, es único y elemental. En cada uno de los atributos sólo se puede incluir un dato, pero no se pueden incluir una lista de datos.

- Segunda Forma Normal o 2FN:

Se dice que un atributo o conjunto de atributos tiene dependencia funcional de otro u otros si a cada uno de los primeros le corresponde sólo uno de los segundos.

---

---





Una tabla está en Segunda Forma Normal o 2FN cuando está en 1FN y todo atributo que no pertenece a la clave primaria tiene una dependencia funcional de la clave completa y no de parte de ella. Luego, si la clave principal está formada por un solo atributo y ya está en 1FN, ya estará en 2FN.

- Tercera Forma Normal o 3FN:

Se dice que hay dependencia funcional transitiva entre dos atributos cuando un atributo que no pertenece a la clave primaria permite conocer el valor de otro atributo.

Una tabla está en Tercera Forma Normal o 3FN si está en 2FN y no existen atributos que no pertenezcan a la clave primaria que puedan ser conocidos mediante otro atributo que no forma parte de la clave primaria, es decir, no hay dependencias funcionales transitivas.

- Forma Normal de Boyce-Codd o FNBC:

Una tabla está en Forma Normal de Boyce-Codd o FNBC si solo existen dependencias funcionales elementales que dependan de la clave primaria o de cualquier clave alternativa. Si la clave primaria está formada por un solo atributo y está en 3FN, ya está en FNBC.

- Cuarta Forma Normal o 4FN:

Existe dependencia funcional multivalorada o de múltiples valores si, dados tres atributos de una tabla, si para cada valor del primer atributo existen múltiples valores en el segundo atributo y no hay ninguna relación entre el tercer atributo y el primero, a no ser a través del segundo atributo.

Una tabla está en Cuarta Forma Normal o 4FN si está en FNBC y las únicas dependencias funcionales multivaloradas que existen son las dependencias funcionales



de la clave con los atributos que no forman parte de la misma. Estas dependencias multievaluadas de la clave con los atributos que no forman parte de la misma son dependencias triviales, por lo que algunos autores dicen que no existen dependencias multievaluadas en 4FN.

- Quinta Forma Normal o 5FN:

Se dice que hay dependencia de JOIN, de unión o de producto si una tabla tiene dependencia de \*unión con varias de sus \*proyecciones y se puede obtener la tabla por medio de la unión de dichas proyecciones. Una tabla esta en Quinta Forma Normal (5FN) o Forma Normal de Proyección-Unión si está en 4FN y las únicas dependencias que existen son las dependencias de unión de una tabla con sus proyecciones relacionándose entre las distintas proyecciones mediante la clave primaria o cualquier clave alternativa. La 5FN se emplea cuando en una misma tabla tenemos mucha información redundante, con pocos atributos o cuando una tabla posee una gran cantidad de atributos y se hace por ello inmanejable.

### **2.1.8 Ventajas de las bases de datos relacionales.**

Las bases de datos relacionales presentan numerosas ventajas:

- Evitan la redundancia de datos. En las bases de datos relacionales, los datos están integrados, por lo que no se almacenan varias copias del mismo dato.
- Consistencia de datos. Al controlar la redundancia, se reduce en gran medida el riesgo de que haya inconsistencia.
- Concurrencia de datos. La base de datos pertenece a la empresa y puede ser utilizada por todos los usuarios autorizados en forma simultánea.
- Mantener estándares. Facilita el intercambio de datos y la administración de los mismos.



- Integridad de datos. A través de reglas que no pueden ser violadas. Las restricciones se aplican tanto a datos, como a sus relaciones.
- El uso de una base de datos relacional bien diseñada puede reducir mucho la cantidad de datos que debe ingresar cada vez que se agrega un registro. Para un número grande de registros, una base de datos relacional puede buscar más rápido entre los mismos.
- Puede crear formularios e informes que muestren solo los datos que se quiere ver y garantiza la integridad referencial, al eliminar un registro todos los registros relacionados dependientes también se eliminan.
- Favorece la normalización por ser más comprensible y aplicable.

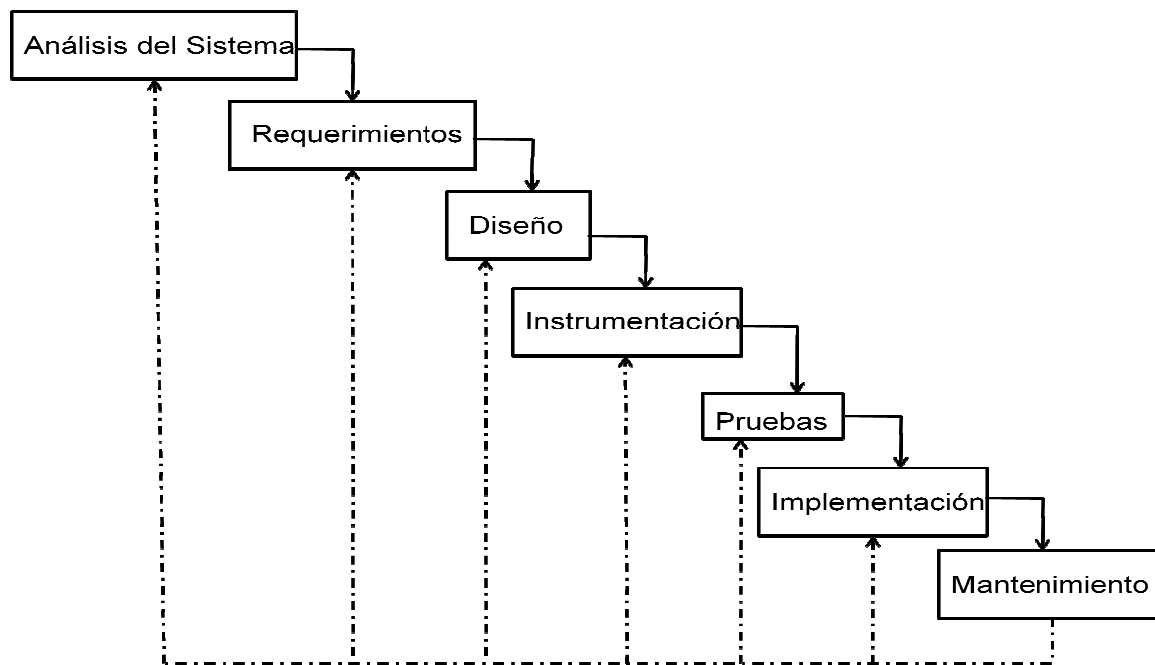
### **2.1.9 Desventajas de las bases de datos relacionales.**

- Se tienen que hacer varios registros, ya que si se tiene una base de datos centralizada, se puede perder la información.
- No se manipulan de forma amigable los bloques de texto como los de tipo dato.
- Presentan deficiencias con datos gráficos, multimedia, CAD (Diseño Asistido por Computadora) y sistemas de información geográfica. Las bases de datos orientadas a objetos (BDOO) se propusieron con el objetivo de satisfacer las necesidades de las aplicaciones anteriores y así, complementar pero no sustituir a las bases de datos relacionales.
- Incompatibilidad. Si las especificaciones se ponen por escrito, no hay problema; pero en la práctica cotidiana, las incompatibilidades mayores o menores entre computadoras, sistemas operativos, lenguajes, protocolos, interfaces y programas de aplicación superan las expectativas. Cuanto más elevado es el número de proveedores, las incompatibilidades son mayores.

## 2.2 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DEL DESARROLLO DEL SISTEMA EN EL MÉTODO DE CICLO DE VIDA CLÁSICO.

El método de Ciclo de Vida para el Desarrollo de Sistemas es el conjunto de actividades que los analistas, diseñadores y usuarios realizan para desarrollar e implantar un sistema de información.

En este método se modela el ciclo convencional de la Ingeniería del Software, aplicando un enfoque sistemático y secuencial de desarrollo. Consta de seis fases que son las siguientes: Análisis del sistema, Diseño, Instrumentación, Pruebas, Implementación y Mantenimiento. Véase Figura 2.2.1.

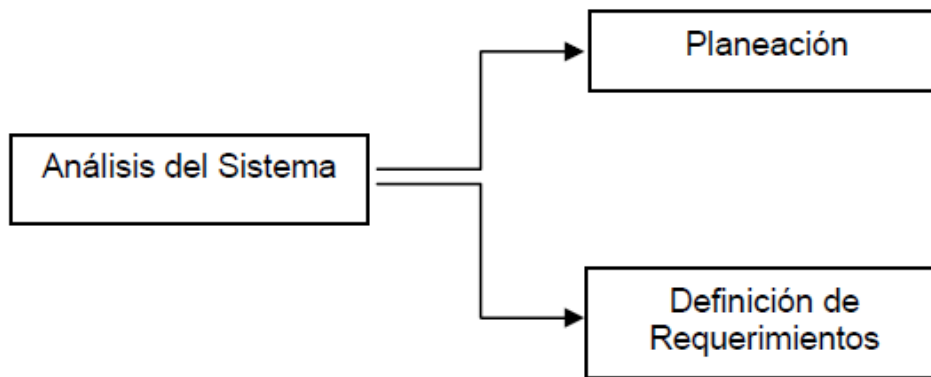


**Figura 2.2.1 Etapas del Método de Ciclo de vida clásico**

Cada fase se efectúa mediante la aplicación de métodos explícitos, herramientas y técnicas. La introducción de una revisión y vuelta atrás, se realiza con el fin de corregir las deficiencias detectadas durante las distintas etapas, o para completar o aumentar las funcionalidades del sistema en desarrollo. De esta manera, durante cualquiera de las fases se puede retroceder momentáneamente a una fase previa para solucionar los problemas que se pudieran haber encontrado.

### 2.2.1 Análisis del Sistema

El Análisis del Sistema se divide en dos subfases: Planeación y Definición de Requerimientos. Véase Figura 2.2.1.1



**Figura 2.2.1.1 División de la Fase de Análisis del Sistema**

En la subfase de Planeación se llevan a cabo las siguientes actividades:

- Comprensión del problema del cliente



Antes de considerar cualquier investigación de sistemas, la solicitud de proyecto debe examinarse para determinar con precisión lo que el solicitante desea antes de seguir adelante, la solicitud de proyecto debe estar claramente planteada.

- Estudio de Factibilidad

Un resultado importante de la investigación preliminar es la determinación de que el sistema solicitado sea factible. En la investigación preliminar existen tres aspectos relacionados con el estudio de factibilidad:

➤ Factibilidad Técnica:

El trabajo para el proyecto, ¿Puede realizarse con el equipo actual, la tecnología existente de software y el personal disponible? Si se necesita nueva tecnología, ¿Cuál es la posibilidad de desarrollarla?

➤ Factibilidad Económica:

Al crear el sistema, ¿los beneficios que se obtienen serán suficientes para aceptar los costos?, ¿los costos asociados con la decisión de *no* crear el sistema son tan grandes que se debe aceptar el proyecto?

➤ Factibilidad Operacional:

Si se desarrolla e implanta, ¿será utilizado el sistema?, ¿existirá cierta resistencia al cambio por parte de los usuarios que dé como resultado una disminución de los posibles beneficios de la aplicación?

- Desarrollo de la estrategia de Solución recomendada
- Determinación de los criterios de aplicación
- Planeación del proceso de desarrollo

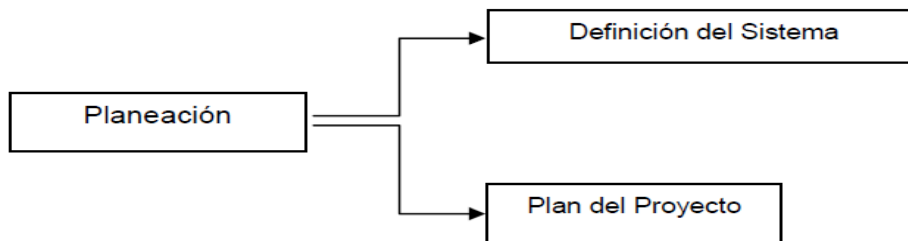


## 2.2.2 Determinación de los requerimientos del sistema

El aspecto fundamental del análisis de sistemas es comprender todas las facetas importantes de la parte de la empresa que se encuentra bajo estudio. Los analistas, al trabajar con los empleados y administradores, deben estudiar los procesos de una empresa para dar respuesta a las siguientes preguntas clave:

- ¿Qué es lo que hace?
- ¿Cómo se hace?
- ¿Con que frecuencia se presenta?
- ¿Qué tan grande es el volumen de transacciones o decisiones?
- ¿Cuál es el grado de eficiencia con el que se efectúan las tareas?
- ¿Existe algún problema? ¿Qué tan serio es? ¿Cuál es la causa que lo origina?

La subfase de Planeación se divide en: Definición del Sistema y Plan de Proyecto. Véase Figura 2.2.2.1



### Figura 2.2.2.1 Productos de la Planeación

La Definición del Sistema puede contener cuadros, figuras, gráficas, y ecuaciones de distintos estilos. La notación exacta empleada en la Definición del Sistema depende mucho del área del problema que se trate. El documento de Definición del Sistema presenta los siguientes elementos:



- Definición del problema.
- Justificación del sistema.
- Metas del sistema y del proyecto.
- Restricciones del sistema y del proyecto.
- Funciones que se proporcionarán (equipo/programación/personal).
- Características del usuario.
- Ambientes de desarrollo/operación/mantenimiento.
- Estrategia de solución.
- Prioridades para las características del sistema.
- Criterios de aceptación del sistema.
- Fuentes de información.
- Glosario de términos.

El Plan del Proyecto contiene el modelo del ciclo de vida que se utilizará, la estructura organizacional del proyecto, la programación preliminar del desarrollo, estimados preliminares de costos y recursos, así como de personal, herramientas y técnicas que se emplearán, y estándares que se seguirán. Los elementos que se deben incluir en el Plan del proyecto son los siguientes:

- Modelo del ciclo de vida (terminología/logros/productos finales).
- Estructura organizacional.
- Estructura de administración/de equipos/distribución de trabajo/definición de puestos
- Requisitos preliminares de personal y recursos.
- Programación preliminar del desarrollo (Redes PERT/Gráficas de Gant).
- Estimado preliminar de costos.
- Mecanismos de supervisión y control del proyecto.





- Herramientas y técnicas que se emplearán.
- Lenguajes de programación.
- Requisitos de prueba.
- Plan de instalación.
- Consideraciones de mantenimiento.
- Método y tiempo de la entrega final.
- Método y tiempo de pago.
- Fuentes de información.

Durante la fase de planeación los estimados de costos y la programación del trabajo serán preliminares, debido a que usualmente no es posible realizar estimaciones precisas sin haber realizado algo del diseño. Las prácticas actuales de contratación requieren que el costo final y la programación se proporcionen durante la fase de planeación. Esta situación, aunada a la naturaleza competitiva del medio, es una de las principales razones de los excesos en costos y las entregas retrasadas de los productos de programación.

Reconociendo esta realidad, muchas organizaciones utilizan una serie sucesiva de estimaciones de costos y programación. Los estimados preliminares se preparan durante la fase de planeación, su redefinición se presenta en la revisión preliminar del diseño; el costo y la programación finales se establecen en la revisión final del diseño. Distintas estimaciones, que representan una clase de capacidades, pueden mostrarse en cada una de las revisiones, de esta manera el cliente y el encargado del desarrollo negociarán un producto para que sea eficiente en términos de costo.

La Definición de Requisitos se refiere a la identificación de las funciones básicas del componente de programación en un sistema. Se pone atención en las funciones y restricciones bajo las cuales se deben de desarrollar.

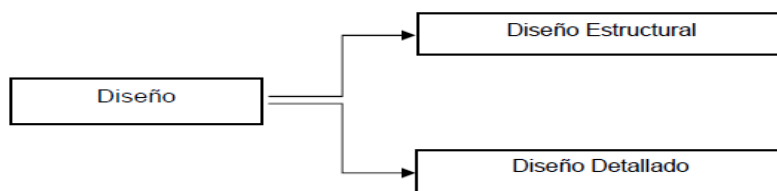


La decisión de cómo se instrumentará la programación se retrasa hasta llegar a la fase de diseño. El producto de la Definición de Requisitos, es una especificación que describe el ambiente de procesamiento, las funciones requeridas de los programas, las restricciones de configuración sobre los programas (tamaño, velocidad, configuración de equipo), manejo de excepciones, subconjuntos y prioridades de instrumentación, cambios probables y modificaciones factibles, así como los criterios de aceptación del producto de programación.

### 2.2.3 Diseño del sistema

Después del Análisis en el modelo de fases, el Diseño de la programación es el siguiente paso. El Diseño de un sistema de información produce los detalles que establecen la forma en la que el sistema cumplirá con los requerimientos identificados durante la fase de análisis. Se refiere a la identificación de los componentes de la programación (funciones, flujo de datos y almacenamiento), especificando las relaciones entre ellos, la estructura de la programación, y manteniendo un registro de las decisiones, proporcionando un documento base para la instrumentación.

El Diseño se divide a su vez en estructural y detallado. Véase Figura 2.2.3.1.



**Figura 2.2.3.1 División de la fase de Diseño.**

El diseño estructural comprende la identificación de los componentes de la programación, su desacoplamiento y descomposición en módulos de procesamiento y estructuras de datos conceptuales, así como la especificación de las interconexiones entre componentes.



El diseño detallado se refiere a detalles de: cómo empacar módulos de procesamiento, y cómo instrumentar los algoritmos, las estructuras de datos y sus interconexiones. Este diseño se relaciona con la adaptación de código existente, modificación de Algoritmos estándar, invención de nuevos algoritmos, diseño de representaciones de datos, e integración del producto final.

El Diseño detallado no es igual que la instrumentación. El primero está muy influido por el lenguaje de programación, pero no tiene que ver con aspectos sintácticos del mismo o con un nivel de detalle como evaluación de expresiones y estatutos de asignación.

#### **2.2.4 Instrumentación**

La fase de Instrumentación en el desarrollo del producto incluye la traducción de las especificaciones del diseño en código fuente, así como su depuración, documentación y pruebas. Los lenguajes de programación modernos proporcionan muchas características para mejorar la calidad del código fuente, como elementos estructurados, tipos de datos predefinidos o definidos por el usuario, verificación de tipos, reglas flexibles de cobertura, mecanismos para manejo de excepciones, elementos concurrentes, y módulos con compilación separada. Algunas de estas características se pueden simular en los lenguajes primitivos, mediante un estilo disciplinado de programación. Los errores descubiertos durante la fase de instrumentación pueden ser errores en las interfaces de datos entre rutinas, errores lógicos en los algoritmos, errores en las estructuras de datos y de falta de consideración de casos de procesamiento. Además, el código fuente puede contener: errores de requisitos, que indican alguna omisión en las necesidades del usuario en el documento de requisitos; errores de diseño, que reflejarán una mala traducción de requisitos en especificaciones y, por último, errores de instrumentación debido a una mala traducción de especificaciones en código fuente.



Una de las metas principales del modelo de fases para el desarrollo de productos de programación es la eliminación de errores de requisitos y diseño, antes de iniciada la instrumentación debido a que es muy caro eliminar errores del análisis y el diseño del código fuente durante la instrumentación y las pruebas.

### **2.2.5 Pruebas**

Las pruebas del sistema comprenden dos tipos de actividades: pruebas de integración y de aceptación. El desarrollo de una estrategia para integrar los componentes de un sistema de programación, en una unidad funcional, requiere una planeación cuidadosa de modo que se disponga de los módulos cuando éstos se necesiten. Las pruebas de aceptación se relacionan con la planeación y ejecución de varios tipos de pruebas para demostrar que el sistema de programación instrumentado satisface las necesidades establecidas en el documento de requisitos.

### **2.2.6 Implementación**

Una vez aceptado por el cliente, el sistema de programación se entrega para su Implantación y operación. Se implementan los niveles de hardware y software que componen el proyecto y el sistema creado en particular. Es en esta etapa cuando realmente se pone a prueba el sistema, para forzarlo y obtener todo el rendimiento posible.

### **2.2.7 Mantenimiento**

Para la fase de Mantenimiento se incluyen actividades como mejoras de las capacidades, adaptación a nuevos ambientes de procesamiento y corrección de fallas del sistema.

### **2.2.8 Ventajas y desventajas del Método de Ciclo de vida clásico.**

#### **Ventajas:**

- Este modelo es el más ampliamente usado por los ingenieros.



- Es muy útil pues ayuda a los desarrolladores a comprender qué es lo que tienen que hacer en cada momento. Su simplicidad hace que resulte sencillo explicárselo a los clientes que no están familiarizados en el proceso software. Además, se muestran de forma explícita qué productos intermedios se tienen que obtener antes de abordar las siguientes tareas.
- El riesgo es bajo para los casos en que se conocen muy bien los requerimientos desde el inicio.
- Es fácil de aplicar.
- Es un modelo sencillo y disciplinado.
- Es fácil aprender a utilizarlo y comprender su funcionamiento.
- Está dirigido por los tipos de documentos y resultados que deben obtenerse al final de cada etapa
- Ha sido muy usado y, por tanto, está ampliamente contrastado.
- Ayuda a detectar errores en las primeras etapas a bajo costo.
- Ayuda a minimizar los gastos de planificación, pues se realiza sin problemas.

Este modelo lineal secuencial es el paradigma más antiguo y más extensamente utilizado en la ingeniería del software. Sin embargo, la crítica del paradigma ha puesto en duda su eficacia.

### **Desventajas**

Las desventajas que presenta el Modelo de Ciclo de Vida Clásico son:

- Los proyectos raramente siguen el proceso lineal tal como se definía originalmente el ciclo de vida.
- Es difícil que el cliente exponga explícitamente todos los requisitos desde el principio.
- El cliente debe tener paciencia. Una versión de trabajo del programa no estará disponible hasta que el proyecto esté muy avanzado.



- Puede resultar complicado regresar a etapas anteriores (ya acabadas) para realizar las correcciones.
- El producto final obtenido puede que no refleje todos los requisitos del usuario.
- Los proyectos reales raras veces siguen la secuencia que propone el modelo. Aunque el modelo lineal puede acoplar interacción, lo hace indirectamente. Como resultado, los cambios pueden causar confusión cuando el equipo del proyecto comienza.
- Se tienen dificultades para hacer cambios en cualquiera de las fases del ciclo.
- La versión funcionando del sistema no estará disponible hasta las etapas finales del desarrollo del proyecto.
- El tiempo que se pasa esperando puede sobrepasar el tiempo que se emplea en el trabajo productivo. Los estados de bloqueo tienden a ser más importantes al principio y al final de un proceso lineal secuencial.

Cada uno de estos errores es real. Sin embargo, el paradigma del ciclo de vida clásico tiene un lugar definido e importante en el trabajo de la ingeniería del software.

A pesar de su antigüedad, el ciclo de vida clásico se ha ganado un lugar importante en el área de la Ingeniería del Software. Proporciona una guía de trabajo en la que se encuentran métodos para el análisis, diseño, codificación, pruebas y mantenimiento. El ciclo de vida en cascada sigue siendo el modelo de proceso más extensamente utilizado por los ingenieros del software, principalmente por su sencillez y facilidad de llevar a cabo. Pese a tener debilidades, es significativamente mejor que un enfoque arbitrario (como el de codificar y corregir) para el desarrollo del software. Muchos de los posteriores modelos de ciclo de vida son, en realidad, modificaciones sobre el modelo clásico, al que se le incorporan nuevas actividades.



## 2.3 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE MICROSOFT ACCESS.

Microsoft Access es un **Data Base Management System (Sistema de Administración de Bases de Datos, DBMS)** clasificado como producto de escritorio, de uso personal o de pequeñas empresas. Es un componente del paquete Microsoft Office. Su función es ser una herramienta de desarrollo de base de datos, capaz de trabajar en si misma o bien con conexión hacia otros lenguajes de programación, tales como Visual Basic 6.0 (VB6) o VisualBasic.NET. Pueden realizarse consultas directas a las tablas contenidas mediante instrucciones SQL. Internamente trae consigo el lenguaje Visual Basic for Application (VBA) el cual es similar en forma a VB6.

De las herramientas rivales del Microsoft Office: IBM Lotus Symphony, Libre Office, Google Docs, Zoho y SoftMaker; solo OpenOffice cuenta con un DBMS, por lo que el Microsoft Access, bien puede decirse, no tiene competencia en el mercado.

### 2.3.1 Ventajas de Microsoft Access

Quizá una de las mayores ventajas que ofrece el Microsoft Access es su capacidad para vincularse con las otras herramientas de la suite Office. Esto permite al usuario interactuar con Word, Excel, PowerPoint y Outlook ya sea recibiendo, enviando información entre ellas. En otras palabras permite el desarrollo de aplicaciones basadas en productos Microsoft. En la figura 2.3.1.1 podemos observar la presentación comercial de Microsoft Access.



**Figura 2.3.1.1 Microsoft Access**

Pero dichas facilidades se extienden a otros productos de Microsoft, como: Visual Basic, MySQL y SQLserver, lo que lo hace más versátil.

Su funcionamiento se basa en un motor llamado Microsoft Jet y permite el desarrollo de pequeñas aplicaciones autónomas formadas por formas Windows y código VBA. Una posibilidad adicional es la de crear archivos con bases de datos que pueden ser consultados por otros programas.

Entre las principales funcionalidades de Microsoft Access se encuentran:

- Crear tablas de datos indexadas.
- Modificar tablas de datos.
- Crear bases de datos relacionales mediante las relaciones entre tablas
- Creación de consultas y vistas.
- Consultas por referencias cruzadas.
- Modificación de bases de datos.
- Formularios.





- Informes.
- Llamadas a la **Application Programation Interface (Interfaz de Programación de Aplicaciones, API)** de Windows.
- Interacción con otras aplicaciones que usen VBA (resto de aplicaciones de Microsoft Office, Autocad, etc.).
- Soporte Unicode: El soporte Unicode de Access permite a las organizaciones multinacionales trabajar con versiones de la aplicación en lenguas diferentes.
- Posibilidad de guardar los archivos en formatos de Access anteriores.
- Autocorrección de nombres: resuelve automáticamente el efecto secundario producido al cambiar el nombre de un objeto de la base de datos.
- Formato condicional: permite utilizar números negativos y positivos, además de valores que pueden expresarse como menor que, mayor que, entre, o igual a.
- Asistente para imprimir tablas relacionadas.
- Agrupación de controles en una sola unidad.
- Compresión automática de la base de datos al cerrar el archivo si la reducción del espacio en disco es significativa.
- Páginas de acceso a datos agrupados, permite a los usuarios extender las aplicaciones de bases de datos a una Intranet corporativa.
- La lista de campos permite agregar información a una vista Página de acceso a datos con sólo arrastrar y colocar los campos.

A continuación se listan algunas de las características técnicas de Microsoft Access, publicadas por el fabricante y que evidencian sus alcances:

- Tamaño del archivo de la base de datos Access: 2GB menos el espacio necesario para los objetos del sistema. (Nota: El tamaño máximo en un archivo sencillo de bases de datos es de 2GB. Se puede trabajar sobre esta limitación enlazándose a tablas de otras bases de datos Access, cada una de las cuales puede ser de hasta 2GB).



- Número de Objetos en una base de datos: 32,768.
- Número de Módulos (Incluyendo formas y reportes que tengan la propiedad HAS Module en Verdadero (True): 1000.
- Número de caracteres en el nombre de un objeto: 64.
- Número de caracteres en un password: 20.
- Número caracteres en un nombre de usuario o grupo: 20.
- Número de usuarios concurrentes: 255.
- Número de tablas abiertas: 2048.
- Tamaño de la tabla: 2GB.
- Número de índices en una tabla: 32.
- Número de campos indexados: 10.
- Tamaño de un campo objeto OLE: 1 GB.

### 2.3.2 Desventajas de Microsoft Access

A continuación se presentan algunas de las desventajas más notables del Microsoft Access:

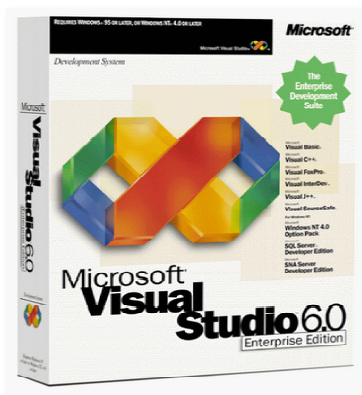
- Problemas a la hora de importar archivos de aplicaciones distintas de Office.
- No es multiplataforma, pues sólo está disponible para sistemas operativos de Microsoft y que no permite transacciones de información con facilidad.
- Para bases de datos de gran tamaño (en cuanto a volumen de datos o de usuarios) es recomendable usar otros sistemas como Microsoft SQL Server, MySQL, Oracle, Informix SQL, ya que no están limitados y permite crear bases de datos más robustas.

## 2.4 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE VISUAL BASIC 6.

### 2.4.1 Origen de Visual Basic.

Visual Basic es un lenguaje de programación dirigido por eventos, este lenguaje de programación es un dialecto de **BASIC (Beginner's All purpose Symbolic Instruction Code (Todo el objetivo del Principiante, Código de Instrucción Simbólico))**, con importantes agregados. Su primera versión fue presentada en 1991, con la intención de simplificar la programación utilizando un ambiente de desarrollo completamente gráfico que facilitara la creación de interfaces gráficas y en cierta medida, también la programación misma.

Es un lenguaje de fácil aprendizaje, guiado por eventos, y centrado en un motor de formularios poderoso que facilita el rápido desarrollo de aplicaciones gráficas. Su principal innovación, que luego fue adoptada por otros lenguajes, fue el uso de un tipo de **DLL (Biblioteca de Enlace Dinámico)**, llamado inicialmente **VBX(Extensión de Visual Basic)** y posteriormente **OCX(Objeto de Vinculación e Integración)**, que permiten contener toda la funcionalidad de un control y facilitar su rápida incorporación a los formularios.



**Fig. 2.4.1.1 Paquetería de Microsoft Visual Studio 6.0 (Visual Basic).**



## 2.4.2 Características Generales

Las aplicaciones creadas con **Visual Basic** se desarrollan partiendo de una interfaz gráfica con la cual es posible generar, de manera automática, conectividad entre controles y datos de manera fácil y rápida mediante la acción de arrastrar y colocar objetos sobre formularios. Los programas quedan compuestos, en una parte, por códigos y, en otra, por partes asociadas a los objetos presentes en la interfaz gráfica. Esto hace que el tipo de programación propio de **Visual Basic** sea parcialmente estructurado y parcialmente orientado a objetos.

**Visual Basic** es un ambiente de programación integrado, esto quiere decir que se utiliza la misma herramienta para editar, compilar, correr, etc. una aplicación. Los procesos se componen de formularios y controles, los cuales tienen propiedades y están diseñados para llevar a cabo procedimientos cuando ocurre algún evento sobre ellos. Además el programa permite usar procedimientos reutilizables que pueden haber sido creados por otras personas o por uno mismo así como módulos de clase para programación orientada a objetos.

**Visual Basic** puede crear componentes **COM (Módulo de Objeto Componente)**, los cuales constituyen la estructura que utiliza Microsoft para todos sus lenguajes actualmente, para desarrollar aplicaciones se utilizan componentes y objetos, parte de esos componentes son los controles ActiveX, servidores Active (DLL o EXE), documentos ActiveX, etc. Cuenta con características para el desarrollo de Internet e Intranet como lo son las aplicaciones **DHTML (Lenguaje Dinámico de Mercado de Hipertexto)**, aplicaciones **IIS (Información de Servicios de Internet)** y soporte para el **FTP (Protocolo de Transferencia de Archivos)**.

La creación de un programa bajo **Visual Basic** requiere de los siguientes pasos:

---

---



- La creación de una interfaz de usuario. Esta interfaz constituirá la vía de comunicación con la máquina para la entrada y salida de datos. Es parte de una ventana o formulario a la que se le van añadiendo los controles pertinentes.
  - La definición de las propiedades de los controles u objetos que hayan sido colocados en el formulario. Estas propiedades determinan la forma estática de los controles, es decir, su forma y sus funciones.
  - La generación de un código asociado a los eventos que ocurran sobre los objetos. A la respuesta a un evento (clic, doble clic, una tecla pulsada, etc.) se le llama procedimiento y deberá desarrollarse de acuerdo al propósito del programa.
  - La generación del código del programa. Un programa puede elaborarse simplemente con la programación de los procedimientos que deban acompañar a cada evento sobre un objeto; sin embargo, **Visual Basic** ofrece la posibilidad de establecer un código de programa separado de estos eventos. Este código puede introducirse en módulos, funciones y procedimientos. Estos procedimientos no responden a un evento acaecido a un objeto, sino que responden a un evento producido durante la ejecución del programa. Visual Basic es extensible gracias a las llamadas al **API (Interfaz de Programación de Aplicaciones)** de Windows, al soporte de terceras partes y a la integración con otras aplicaciones de Windows. Es utilizado principalmente para aplicaciones de gestión de empresas, debido a la rapidez con la que puede hacerse un programa que utilice una base de datos sencilla, dado que cuenta con varias bibliotecas para el manejo y acceso a bases de datos, entre las que destaca ADO, Jet, ODBC y RDO; programadores en este lenguaje.
  - Finalmente, las numerosas bibliotecas **DLL (Biblioteca de Enlace Dinámico)** facilitan el acceso a muchas funciones del sistema operativo y la integración con otras aplicaciones. **Visual Basic** tiene también sus lenguajes derivados como lo son el VBScript y el Visual Basic for Applications VBA, el primero es el lenguaje predeterminado para Active Server Pages ASP y el segundo es más bien una
- 
-



extensión del mismo lenguaje, el cual permite codificar módulos o macros que se utilizan en los programas de Microsoft Office.

En la actualidad existe una nueva variante del lenguaje que se llama Visual Basic.NET, el cual es un programa que cuenta ya con funciones equivalentes a C++ y con una orientación a objetos como la de JAVA.

### 2.4.3 Ventajas y Desventajas de Visual Basic.

#### Ventajas

- Visual Basic es un lenguaje simple y, por tanto, más fácil de aprender que otros lenguajes más complejos como DELPHI y Power Builder, entre otros. Su mayor simplicidad radica en el desarrollo de formularios, mediante el arrastre de controles. Es un lenguaje compatible con Microsoft Office y otros lenguajes (Informix, Oracle, Microsoft SQL, etc).
  - Gran parte del trabajo en el diseño de formularios está realizado, gracias a la gran gama de controles incorporados junto al lenguaje, cuyas propiedades y métodos son fáciles de manipular, lo cual, junto con todas las demás características del programa, hace que el desarrollo de aplicaciones se vuelva sumamente rápido, permite crear controles personalizados fácilmente del mismo modo que el diseño de los formularios.
  - Puede crear controles ActiveX más fácilmente que si se usa C++, cuenta con un excelente paquete de ayuda, cuenta con herramientas para el desarrollo integración de Internet/Intranet, se pueden crear servidores fácilmente. Es excelente para cálculos intensivos del **CPU (Unidad Central de Procesos)** como por ejemplo operaciones matemáticas.
- 
-



## Desventajas

- El dueño de Visual Basic es Microsoft, por lo tanto nadie que no sea del equipo de desarrollo de esta compañía decide la evolución del lenguaje.
  - Sólo existe un compilador, llamado igual que el lenguaje, generando ejecutables para Windows, la sintaxis es bastante inflexible, los ejecutables generados son relativamente lentos, no es adecuado para aplicaciones grandes, multimedia, de oficina, videojuegos, editores gráficos, etc.
  - No cuenta con características para programación avanzada, no permite generar librerías dinámicas (DLL), sólo permite el uso de funciones de librerías dinámicas (DLL) stdcall. para que los ejecutables que generan funciones, necesita una DLL llamada MSVBVMxy.DLL: MicroSoft Visual Basic Virtual Machine x.y.
  - Algunas funcionalidades están indocumentadas, la ligera implementación de la OOP no permite sacar el máximo provecho de este modelo de programación.
  - No soporta el tratamiento de procesos como parte del lenguaje, no maneja excepciones, no incluye operadores a nivel de bits, no contempla el manejo de memoria dinámica, punteros, arrays, etc. como parte del lenguaje.
  - No puede avisar ni advertir cuando están presentes ciertos errores, como sería una inadecuada conversión de tipos.
- 
-



- El tratamiento de mensajes de Windows es básico e indirecto; la gran gama de controles incorporados son, sin embargo, muy generales, lo cual lleva a tener que reprogramar nuevos controles para una necesidad particular de la aplicación.
  
- Los controles personalizados no mejoran la potencia de la API de Windows y, en algunos casos, acudir a ésta es la única manera de conseguir el control personalizado deseado.
- No tiene la misma funcionalidad que C++ a la hora de obtener características de bajo nivel del sistema operativo. La forma de programación que plantea Visual Basic ha ocasionado que muchos programadores de este lenguaje practiquen malos hábitos, entre los que se encuentran:
  - La utilización de variables globales.
  - Omitir la declaración de variables.
  - Trabajar con variables de tipo indefinido (Tipo Variant).
  - Escribir código innecesario.
  - Escribir código repetido a falta de funciones y procedimientos.
  - El uso incorrecto de la API de Windows.
  - El uso de goto y etiquetas.
  - El uso de controles como simples contenedores de datos.
  - El crear dependencia de los controles a la hora de programar.





## **2.5 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE LA ARQUITECTURA CLIENTE-SERVIDOR.**

La arquitectura cliente-servidor es una técnica para dividir y especializar programas y equipos de cómputo, a fin de que la tarea que cada uno de ellos realiza se efectúe con la mayor eficiencia y permita simplificarlas.

En esta arquitectura la capacidad de proceso está repartida entre el servidor y los clientes. Los clientes interactúan con el usuario, usualmente en forma gráfica. Frecuentemente se comunican con procesos auxiliares que se encargan de establecer conexión con el servidor, enviar el pedido, recibir la respuesta, manejar las fallas y realizar actividades de sincronización y de seguridad.

Los servidores proporcionan un servicio al cliente y devuelven los resultados. En algunos casos existen procesos auxiliares que se encargan de recibir las solicitudes del cliente, verificar la protección, activar un proceso servidor para satisfacer el pedido, recibir su respuesta y enviarla al cliente. Además deben manejar los interbloques, la recuperación ante fallas, y otros aspectos afines.

Para que los clientes y los servidores puedan comunicarse se requiere una infraestructura de comunicaciones, que proporcione los mecanismos básicos de direccionamiento y transporte.

### **2.5.1 Características del Modelo Cliente-Servidor.**

- El Cliente y el Servidor pueden actuar como una sola entidad y también pueden actuar como entidades separadas, realizando actividades o tareas independientes.
- Las funciones de Cliente y Servidor pueden estar en plataformas separadas o en la misma plataforma.



- Un servidor da servicio a múltiples clientes.
- Cada plataforma puede ser escalable independientemente. Los cambios realizados en las plataformas de los Clientes o de los Servidores, ya sean por actualización o por reemplazo tecnológico, se realizan de una manera transparente para el usuario final.

El Cliente actúa de la siguiente forma:

- Establece una conexión con el servidor (Crea un socket con el servidor).
- Manda mensajes al servidor o espera un mensaje de él (Consultas).
- Repite el paso anterior mientras sea necesario.
- Cerrar la conexión con el servidor.

El Servidor actúa de la siguiente manera:

- Inicializa un puerto de comunicación, en espera de clientes que intenten conectarse a él (Crea un serverSocket).
- Una vez que se conecta alguien, crea un **thread (hilo)** de ejecución para este usuario mientras que el thread principal vuelve al paso anterior. Esto comúnmente se hace para que el servidor pueda atender a varios clientes al mismo tiempo.
- Se comunica con el cliente mediante el socket creado entre el cliente y él.
- Espera que el cliente se vaya o lo desconecta el mismo servidor (Cierra el socket entre ellos) y elimina el thread de comunicación entre ellos.

## 2.5.2 Clasificación por capas de la arquitectura Cliente-Servidor.

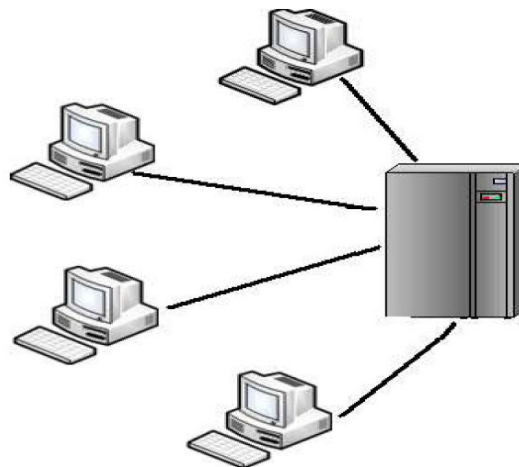
### 2.5.2.1 Modelo Cliente-Servidor de una capa.

Las aplicaciones cliente-servidor de una capa agrupan la lógica de presentación (interfaz de usuario), la lógica de aplicación y la fuente de datos (base de datos) en una sola máquina, la cual actúa simultáneamente como cliente y servidor.

### 2.5.2.2 Modelo Cliente-Servidor de dos capas.

Tradicionalmente la arquitectura Cliente-Servidor está basada en un modelo de computación de dos capas. Este modelo consiste de un cliente y un servidor de base de datos. El procesamiento de tareas y la lógica de la aplicación son compartidas entre el servidor de base de datos y el cliente.

Como podemos observar en la figura 2.5.2.2.1, en este modelo a los clientes se les llama clientes pesados, en donde reside mucho del poder de procesamiento y de la lógica de la aplicación. Esto hace que el mantenimiento del cliente sea costoso. Adicionalmente, los clientes pueden estar operando en diferentes plataformas, dando como resultado una distribución compleja de versiones de aplicaciones específicas de las plataformas.



**Figura 2.5.2.2.1 – Arquitectura Cliente Servidor de 2 capas**

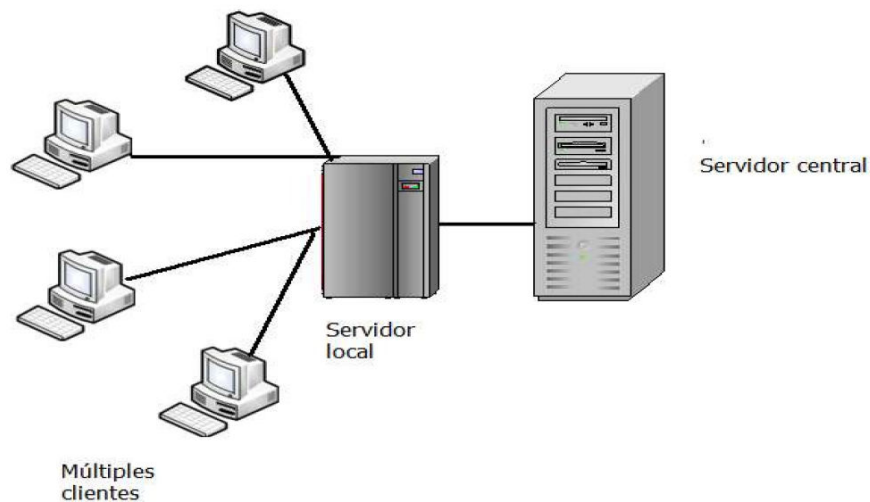
### 2.5.2.3 Modelo Cliente-Servidor de tres capas.

El modelo de computación de tres capas es un modelo evolucionado del modelo de dos capas. En este modelo, existe una capa intermedia entre el cliente y el servidor de base

---

de datos. Esta capa consiste de un servidor de aplicaciones que contiene el grueso de la lógica de la aplicación. Los clientes en este modelo son clientes livianos o clientes ligeros. Con esta arquitectura la lógica de la aplicación reside en una sola capa que puede ser fácilmente mantenida. El diseño arquitectónico de la capa media puede también ser optimizada en funciones del servidor puesto que éste no tiene que contener u hospedar la base de datos.

En esta arquitectura de tres capas, el software del cliente (capa cliente) es ligero, suficiente para ser descargado bajo demanda y lo suficientemente pequeño como para presentar la interfaz del usuario. El grueso de la lógica de la aplicación está implementada ya sea en la capa media (servidor de aplicaciones) o está almacenada en la base de datos, como podemos observar en la figura 2.5.2.3.1.



**Figura 2.5.2.3.1 – Arquitectura Cliente Servidor de 3 capas**



#### 2.5.2.4 Modelo Cliente-Servidor de n capas.

El despliegue de una plataforma capaz de soportar los requerimientos de aplicaciones en Internet, invariablemente requieren de una arquitectura de alto nivel de múltiples capas.

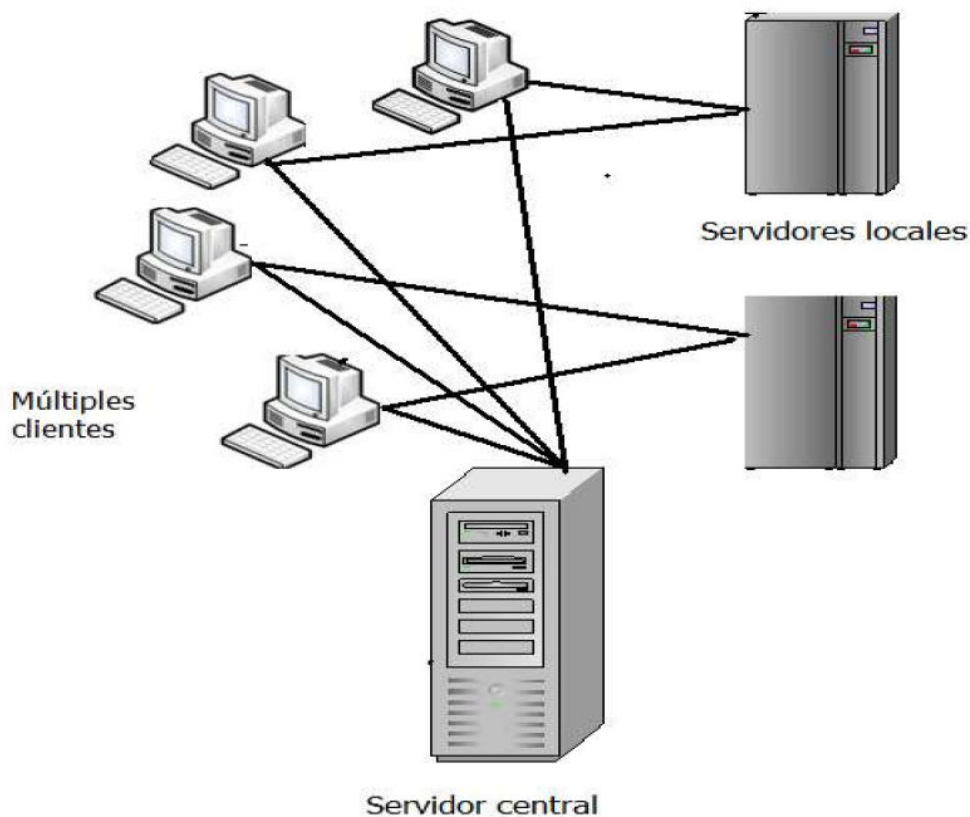
En la capa cliente, los clientes con un navegador basado en HTML soportan los requerimientos de accesibilidad **anytime (cuando sea)** y **anywhere (donde sea)**: **GUIs (Graphical User Interface, Interfaz grafica de usuario)** más complejas, pueden ser soportados dentro del navegador descargando más código de la aplicación cliente, descansando en **plugins (Accesorios)** o en capacidades específicas del navegador.

La capa del servidor de presentación consiste de servidores WEB que manejan requerimientos http y genera dinámicamente código de presentación para su ejecución y despliegue en el cliente, de forma típica en la forma de páginas HTML. Muchos de los requerimientos de personalización son manejados en esta capa. El servidor de presentación puede parametrizar la generación de código de presentación del cliente basándose en el **ID (Identifier –Identificador)** del usuario, preferencias, roles, afiliación, etc.

Los servidores dan cabida a la lógica de la aplicación que idealmente puede ser reutilizada a través de una variedad de clientes y otras aplicaciones. La lógica de la aplicación también debería poder ser llamada fácilmente desde clientes y servidores externos para facilitar los requerimientos de integración de ésta. La capa del servicio de la arquitectura también contiene servidores especializados de análisis y reportes para complementar la aplicación.

Finalmente, la capa de servidores de datos converge hacia una implementación estándar suficiente para satisfacer las demandas de almacenamiento, manipulación,

recuperación y análisis de requerimientos de aplicaciones en Internet. Esta capa también contiene fuentes de datos externas y aplicaciones que deben estar integradas en el sistema. En la figura 2.5.2.4.1 podemos observar una representación de la interacción lógica de este modelo de la arquitectura.



**Figura 2.5.2.4.1 - Arquitectura Cliente Servidor de n capas**

## 2.5.3 Elementos de red dentro de la arquitectura Cliente-Servidor.

### 2.5.3.1 Hub

Un **Hub (Concentrador)** es un equipo de conexión de redes, que funciona como repetidor de la señal eléctrica. Es decir toma la señal que le llega por un puerto y la

---



reproduce íntegramente por los demás. Nada de comprobación de tramas ni toma de decisiones. El ancho de banda es compartido por todos los equipos conectados a él.

### 2.5.3.2 Bridge

Un **bridge (Puente)** es un dispositivo de interconexión de redes que opera en la capa 2 (nivel de enlace de datos) del modelo OSI. Este interconecta dos segmentos de red (o divide una red en segmentos) estableciendo el paso de datos de una red hacia otra, con base en la dirección física de destino de cada paquete.

Funciona a través de una tabla de direcciones **MAC (Media Access Control- Control de Acceso al Medio)** detectadas en cada segmento a que está conectado. Cuando detecta que un nodo de uno de los segmentos está intentando transmitir datos a un nodo del otro, el bridge copia la trama para la otra subred.

La principal diferencia entre un bridge y un hub es que el segundo pasa cualquier trama con cualquier destino para todos los otros nodos conectados, en cambio el primero sólo pasa las tramas pertenecientes a cada segmento. Esta característica mejora el rendimiento de las redes al disminuir el tráfico inútil.

### 2.5.3.3 Switch

El **switch (Conmutador)** es un dispositivo muy similar al bridge, opera en la misma capa del modelo OSI y tiene un comportamiento similar, con la diferencia de enlazar más de 2 segmentos de red, pueden ser 3 o más. Opera en forma similar, mediante tablas de dirección MAC, en donde cada puerto contiene una lista de las direcciones de destino que son alcanzables dentro del mismo.

---

---



#### 2.5.3.4 Router

Un **router (Ruetador)** es un dispositivo hardware o software (vrouter –virtual router) de interconexión de redes de computadoras que opera en la capa tres (nivel de red) del modelo OSI. Este dispositivo interconecta segmentos de red o redes enteras. Hace pasar paquetes de datos entre redes tomando como base la información de la capa de red.

El router toma decisiones lógicas con respecto a la mejor ruta para el envío de datos a través de una red interconectada y luego dirige los paquetes hacia el segmento y el puerto de salida adecuados. Sus decisiones se basan en diversos parámetros. Una de los más importantes es decidir la dirección de la red hacia la que va destinado el paquete (En el caso del protocolo **IP (Internet Protocol)**, esta sería la dirección IP). Otras decisiones son la carga de tráfico de red en las distintas interfaces de red del router y establecer la velocidad de cada uno de ellos, dependiendo del protocolo que se utilice.





**CAPÍTULO 3**  
**ANÁLISIS Y PLANTEAMIENTO**  
**DEL PROBLEMA**



### 3.1 IDENTIFICACIÓN DEL PROBLEMA

La Sociedad de Ahorro es una compañía autorizada por el gobierno para captar dinero a cambio de un interés. El producto que la sociedad ofrece es un servicio a una persona física o moral a través de una cuenta de ahorro, a la cual se agrega un interés mensual, trimestral, anual o por la vida del contrato.

Los movimientos del cliente se deben de reportar consistentemente, en función de los movimientos realizados, los intereses devengados y los cobros extraordinarios que se hacen por infracciones al contrato del cliente (recargos, intereses moratorios, comisiones, etc.), se asocian a cuentas concentradoras que la CNBV (Comisión Nacional Bancaria y de Valores), supervisa para así poder mantener dentro de un marco legal el funcionamiento de la institución bancaria.

Cabe señalar que el ahorro de los cuentahabientes, es una vía de capitalización para la Sociedad de Ahorro. El capital es utilizado por ésta para diversos fines, desde la inversión por lucro hasta para cumplir con los intereses de sus cuentahabientes.

Como se mencionó anteriormente, el proceso de la administración de las cuentas de ahorro se hace de forma manual y del que a continuación se hace una breve descripción para posteriormente describir el proceso una vez implementado el sistema de administración y control de cuentas de ahorro.



- El proceso inicia con la asistencia del cliente a la Sociedad de Ahorro. El cliente asiste por cualquiera de las tres razones siguientes:
    - Desea abrir una cuenta de ahorro/a plazos/Ahorro fácil.
    - Realizar un movimiento: depósito/retiro
    - Aclaración
  
  - Si desea abrir una cuenta se sigue el siguiente proceso:
    - En el área de atención al cliente solicita la apertura de una cuenta de ahorro/a plazos/monedero electrónico.
    - El subgerente o el ejecutivo de cuenta le describe los productos que para tal efecto promueve la Sociedad de Ahorro.
    - Una vez que el cliente eligió el servicio de su conveniencia, entrega su identificación a fin de comprobar su identidad y residencia. Se le toman datos generales, que se vacían en el contrato correspondiente. Se firma y se le entrega copia del mismo, así como su libreta de ahorro con su número de cuenta correspondiente. El cliente apertura ésta con la cantidad establecida por la institución para tal caso. El cliente se retira.
    - La información, en copia, pasa a la oficina correspondiente a fin de que se le abra expediente. La información es capturada en Excel y almacenada.
  
  - El cuentahabiente desea realizar un movimiento en su cuenta:
    - Para depósito y/o retiro el cuentahabiente llena ficha con la cantidad del movimiento (efectivo/cheque), su nombre y número de cuenta. Presenta esta al cajero, junto con su identificación –si es retiro- y su libreta de
- 
-



ahorro.

- El cajero efectúa el movimiento, declara el movimiento en la libreta de ahorro, la sella y la firma entregando la libreta, el efectivo y la identificación –si fue retiro- al cliente.
  - Se transcriben los datos del movimiento a un archivo Excel.
  - El cliente se retira.
  - La ficha se almacena para enviarla al área de administración a fin de mes.
- El cuentahabiente desea hacer una aclaración:
    - El cuentahabiente pasa al área de atención al cliente donde manifiesta su inquietud al asesor de cuenta o subgerente, presentando su libreta de ahorro, estado de cuenta e identificación.
    - El asesor en función de los datos del estado de cuenta y la libreta de ahorro determina si la inquietud del cliente es fundada o no.
    - Si la inquietud tiene fundamento, el asesor procede a levantar un reporte para que el área de administración lleve a cabo la corrección correspondiente. Si la inquietud no tiene fundamento, el asesor realiza la aclaración correspondiente. El cuentahabiente se retira.
  - El área de administración recibe la información en archivos Excel, para su procesamiento. Este tiene un doble propósito: cumplir con las obligaciones legales de entregar reporte a la CNBV y generar estado de cuenta para los cuentahabientes. En ambos casos se cuenta con un área para el proceso de la información. Al proceso que realiza la sociedad de ahorro para llevar el control de la administración los ingresos que entran y salen se denomina “Conciliación”. La documentación que se genera a partir de los movimientos en las cuentas es la que pasa a la central de la sociedad para su conciliación.
  - La información actualizada es respaldada en el servidor de la Sociedad de
-



Ahorro.

- Se imprimen los estados de cuenta y se envían a los cuentahabientes.

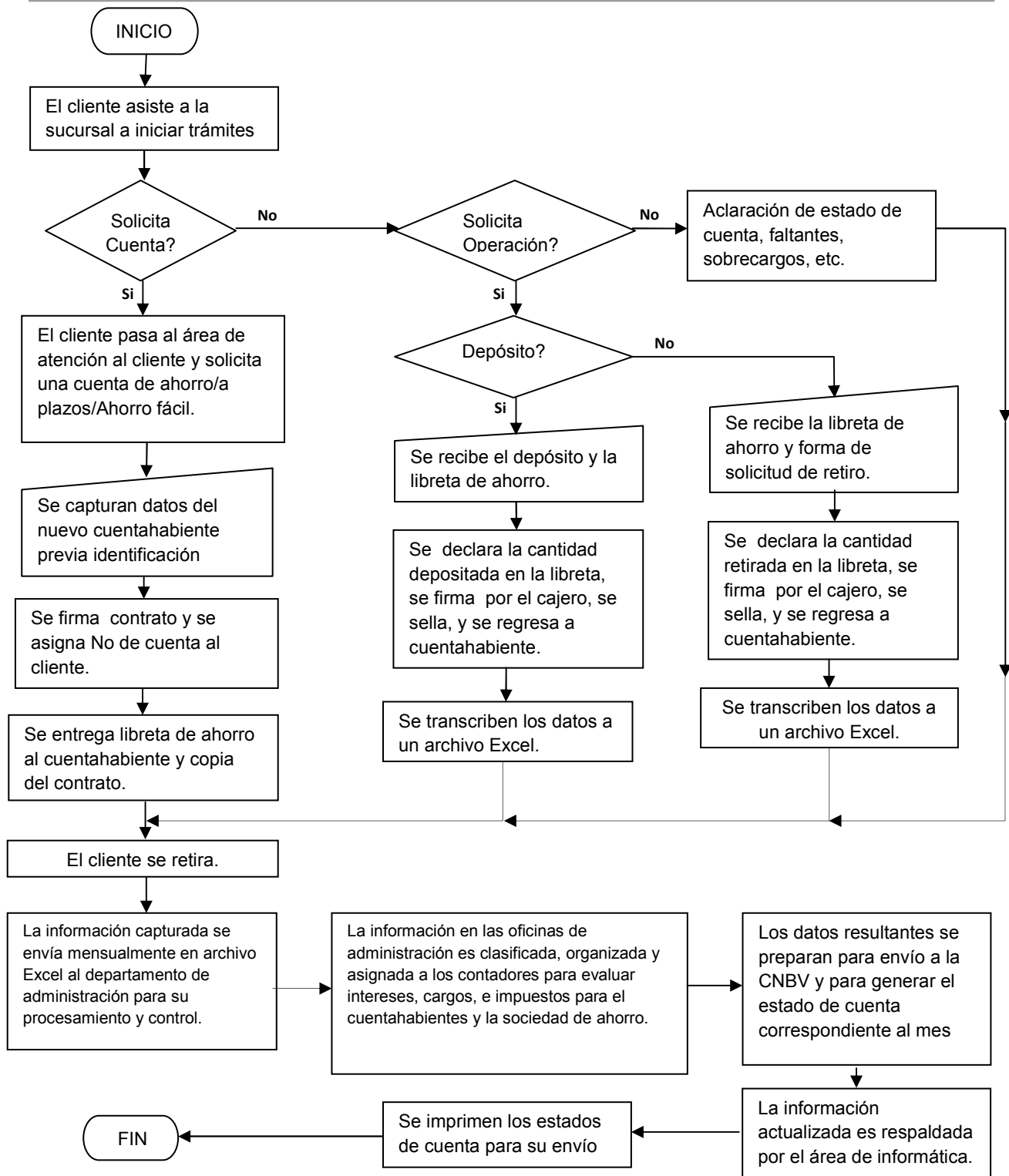
El proceso descrito deja ver que la operación manual es esencial en este y se deduce que la probabilidad de error esta siempre latente, debido a la gran cantidad de personas que intervienen en él.

La atención al cliente es lenta derivada del proceso manual, tanto para activar una cuenta como para declarar los movimientos de las mismas, lo que repercute en una mala imagen para la Sociedad de Ahorro. A lo anterior hay que añadir el tiempo perdido por transcribir el movimiento en planilla en Excel.

De lo anterior se deduce, que los costos de operación se derivan en buena parte por la mano de obra, la papelería y otros gastos.

La implementación del sistema reducirá los gastos en esos rubros, al contar con la tecnología que permita a la Sociedad de Ahorro atender un mayor número de clientes con mano de obra mínima y menor cantidad de insumos.

En la figura 3.1.1 se muestra el esquema que el proceso que lleva actualmente.



**Figura 3.1.1. Diagrama del Modelo actual de operación de la Sociedad de Ahorro.**



## **3.2 RECOPIACIÓN Y ANÁLISIS DE LA INFORMACIÓN.**

Actualmente la Sociedad de Ahorro recopila, analiza, genera reportes y almacena toda la información de las operaciones realizadas, las cuentas y los clientes en archivos Excel. Esto conlleva un sin número de desventajas, entre las principales la falta de integridad referencial y la vulnerabilidad a la pérdida de la misma.

A continuación, describiremos los distintos procesos seguidos por la Sociedad de Ahorro. Dichas descripciones contendrán muestras de los distintos formatos involucrados en dicho proceso, así como menciones a los actores involucrados en los mismos.

### **3.2.1 Alta de cliente nuevo.**

En éste proceso, se considera que un individuo con la necesidad y los medios de convertirse en cliente la Sociedad de Ahorro, se ha aproximado a la misma con la intención de contratar por lo menos un producto. Dicho cliente, potencial, deberá entregar la documentación mínima necesaria (requerida por los organismos legales que rigen la operación de las instituciones financieras en nuestro país), así como llenar una solicitud en la que se especifique el o los servicios que desea contratar. Hasta este momento, no se cuentan con productos contratados, solo se realiza el alta de cliente nuevo, en donde éste recibirá un número de cliente, y la Sociedad de Ahorro contará con la información necesaria para el correcto manejo de los futuros productos

---

---



contratados. En la figura 3.2.1.1 podemos observar el actual formato para dar de alta un cliente nuevo. Debemos mencionar que actualmente, si el cliente decide actualizar sus datos, se debe llenar un nuevo formato de alta de cliente nuevo, y anexarse al original, indicando la leyenda “ACTUALIZACION” en el encabezado del mismo.



**Sociedad de Ahorro**  
... tu patrimonio seguro

## Sociedad de Ahorro

### Contrato para cliente nuevo

Sucursal: Fecha:  /  /

**Datos del Cliente**

**Nombre**

**Domicilio**

<i>Calle</i>	<i>No Ext</i>	<i>No Int</i>	<i>Colonia</i>
<i>Delegación o Municipio</i>	<i>Entidad Federativa</i>		<i>Código Postal</i>
<i>Teléfono</i>	<i>RFC</i>	<i>Sexo</i>	

**Referencias**

	<i>Nombre</i>	<i>Teléfono</i>
1	<input type="text"/>	<input type="text"/>
2	<input type="text"/>	<input type="text"/>
3	<input type="text"/>	<input type="text"/>

**Productos a contratar:**

Ahorra Fácil	<input type="text"/>	Pesos	<input type="text"/>	Dolares	<input type="text"/>	Euros	<input type="text"/>
Ahorro Normal	<input type="text"/>	Pesos	<input type="text"/>	Dolares	<input type="text"/>	Euros	<input type="text"/>
A Plazos	<input type="text"/>	Pesos	<input type="text"/>	Dolares	<input type="text"/>	Euros	<input type="text"/>
Cuenta Corriente	<input type="text"/>	Pesos	<input type="text"/>	Dolares	<input type="text"/>	Euros	<input type="text"/>

**Para Uso del Banco**

Numero de Cliente:

---

*Representante del Banco*

---

*El cliente*

Figura 3.2.1.1 Formato de alta de cliente nuevo.





### **3.2.2 Apertura de una cuenta nueva.**

Una vez que el cliente en potencia se ha convertido en cuentahabiente, es decir, la Sociedad de Ahorro tiene la documentación que satisface los requerimientos legales pertinentes y éste cuenta con un número de cliente, se procede el alta de los productos a contratar por el interesado. Durante éste proceso, el cliente debe firmar una copia del contrato del producto, donde se especifican los derechos y obligaciones de ambas partes celebrantes, cliente y la Sociedad de Ahorro.

El cliente, de igual forma, deberá realizar las transacciones financieras iniciales cumpliendo cuando menos con los montos mínimos establecidos en el contrato, debiendo anexarse una copia del comprobante de dicha operación de depósito en ventanilla al expediente. Si no se cuenta con cualquiera de estos requisitos y de acuerdo a lo especificado en el contrato, el cliente será sujeto a sanciones ó incluso a la cancelación del producto. En la figura 3.2.2.1 podemos observar el formato empleado para la apertura de una cuenta nueva.





**Sociedad de Ahorro**  
... tu patrimonio seguro

## Sociedad de Ahorro

### Contrato para cuenta nueva

Utilice un formato por cada cuenta

Sucursal:  Fecha  /  /

**Datos del Cliente**

Nombre

Numero de Cliente:

Productos a contratar:

Ahorra Facil <input type="checkbox"/>	Pesos <input type="checkbox"/>	Dolares <input type="checkbox"/>	Euros <input type="checkbox"/>
Ahorro Normal <input type="checkbox"/>	Pesos <input type="checkbox"/>	Dolares <input type="checkbox"/>	Euros <input type="checkbox"/>
A Plazos <input type="checkbox"/>	Pesos <input type="checkbox"/>	Dolares <input type="checkbox"/>	Euros <input type="checkbox"/>
Cuenta Corriente <input type="checkbox"/>	Pesos <input type="checkbox"/>	Dolares <input type="checkbox"/>	Euros <input type="checkbox"/>

**Para Uso del Banco**

Numero de Cuenta

Saldo Inicial \$

Ejecutivo de Cuenta:

---

*Representante del Banco*

---

*El cliente*

Figura 3.2.2.1 Formato de apertura de una cuenta nueva.

### 3.2.3 Cancelación de una cuenta.

Éste proceso, donde se considera que el producto contratado por el cliente ha concluido, y se detienen todas las obligaciones entre las partes involucradas, debiendo ambas no tener adeudos con el segundo, es decir, el cliente no adeudar comisiones o



algún tipo de pago y la Sociedad de Ahorro haber entregado la totalidad del saldo del producto al cliente. La figura 3.2.3.1 ejemplifica el formato empleado para dicha función.



**Sociedad de Ahorro**  
... tu patrimonio seguro

# Sociedad de Ahorro

## Cancelacion de cuenta

Sucursal:  Fecha  /  /

**Datos del Cliente**

Nombre

Numero de Cliente:

Producto a cancelar:

Ahorra Facil	<input type="checkbox"/>	Pesos	<input type="checkbox"/>	Dolares	<input type="checkbox"/>	Euros	<input type="checkbox"/>
Ahorro Normal	<input type="checkbox"/>	Pesos	<input type="checkbox"/>	Dolares	<input type="checkbox"/>	Euros	<input type="checkbox"/>
A Plazos	<input type="checkbox"/>	Pesos	<input type="checkbox"/>	Dolares	<input type="checkbox"/>	Euros	<input type="checkbox"/>
Cuenta Corriente	<input type="checkbox"/>	Pesos	<input type="checkbox"/>	Dolares	<input type="checkbox"/>	Euros	<input type="checkbox"/>

**Para uso del Banco**

Numero de Cuenta

Ejecutivo de Cuenta

---

*Representante del Banco*

---

*El cliente*

Figura 3.2.3.1 Cancelación de cuenta.



### 3.2.4 Depósito en ventanilla.

En éste proceso, se genera un comprobante al cliente que ampara la cantidad depositada a su cuenta. Dicho comprobante deberá marcar fecha de la transacción, el número de cliente, el número de cuenta, la cantidad depositada. Además de generarse por duplicado, quedando una copia en el archivo de la Sociedad de Ahorro y otra entregada al cliente. En la figura 3.2.4.1 se puede observar la forma para dicha operación.



**Sociedad de Ahorro**  
... tu patrimonio seguro

## Sociedad de Ahorro

### Deposito en Efectivo

Fecha

*Datos del Cliente*

Nombre

Numero de Cuenta:

Cantidad a Depositar: \$

Importe con Letra

---

*Sello y Firma del Cajero*

---

*El cliente*

Cajero:

Numero de Operacion:

Numero de Cliente:

Figura 3.2.4.1 Depósito en efectivo.



### 3.2.5 Retiro en ventanilla.

En éste proceso, se genera un comprobante al cliente que ampara la cantidad retirada a su cuenta. Dicho comprobante deberá marcar fecha de la transacción, el nombre de cliente, el número de cuenta, la cantidad retirada. Además de generarse por duplicado, quedando una copia en el archivo de la Sociedad de Ahorro y otra entregada al cliente.



**Sociedad de Ahorro**  
... tu patrimonio seguro

## Sociedad de Ahorro

### Retiro de Efectivo

Fecha  /  /

*Datos del Cliente*

Nombre

Numero de Cuenta:

Cantidad a Retirar: \$

Importe con Letra

---

*Sello y Firma del Cajero*

Cajero:

---

*El cliente*

Numero de Operacion:

Numero de Cliente:

Figura 3.2.5.1 Retiro de Efectivo.



### 3.2.6 Reporte mensual de movimientos por cuenta.

Este es un reporte que se conoce comúnmente como estado de cuenta, en el que se detallan en forma consecutiva todas las transacciones realizadas durante un periodo de un mes en cada cuenta. Cuenta con toda la información del cliente, así como los datos de la cuenta.

Para generar este reporte, se toma el reporte de corte de caja, y se transcribe en forma consecutiva las operaciones, tomando el número de operación, la fecha, la hora, la descripción y el monto de la misma. Al finalizar el mes, se calcula el saldo de la cuenta, finalizando el reporte. La figura 3.2.6.1 muestra un reporte mensual de movimientos sin llenar.

The image shows a form titled "Reporte mensual de movimientos" from "Sociedad de Ahorro". The form includes fields for: "Fecha" (Date), "<Tipo de Cuenta>" (Account Type), "Nombre" (Name), "Numero de Cliente:" (Client Number), "Periodo del:" (Period from) and "at:" (to), "Numero de Cuenta:" (Account Number), "Saldo Inicial del periodo:" (Initial Balance), and "Saldo al corte:" (Balance at cut-off). Below these fields is a section titled "Detalle de movimientos" which contains a table with the following columns: "Fecha", "Descripcion", "Cargo", "Abono", and "Saldo". The table is currently empty.

Figura 3.2.6.1 Reporte mensual de movimientos por cuenta.



### **3.2.7 Corte de caja diario.**

Este es un registro que llena el cajero en forma consecutiva de acuerdo a las operaciones que realiza cada día. Debe registrar los números de operación, cliente, cuenta, una descripción de la operación, el monto de la misma, así como la hora en la que se realiza. Este reporte se concluye al finalizar el horario de sucursal para la actualización de las cuentas.

### **3.2.8 Reporte variable de saldos en caja por sucursal.**

Este es un reporte muy similar al corte de caja diario, solo que generado con una periodicidad semanal, quincenal o mensual, y detalla los saldos en caja disponibles a la Sociedad de Ahorro. El nivel de detalle es diferente, ya que en lugar de operaciones consecutivas, está compuesto de los totales del reporte de corte diario a caja.



### **3.3 LEVANTAMIENTO DE LOS REQUERIMIENTOS GENERALES Y PARTICULARES QUE DEBE CUBRIR LA SOLUCIÓN DE SOFTWARE A CONSTRUIR.**

El objetivo de este proceso es comprender en profundidad la necesidad de nuestro cliente, el problema que desea resolver, con la finalidad de ofrecer un sistema óptimo que se adapte a sus necesidades, obedeciendo también a los recursos con los que se cuenta.

#### **3.3.1 Requerimientos generales**

Actualmente la Sociedad de Ahorro requiere de un sistema que le permita llevar de forma eficiente el control y la administración de las cuentas de ahorro de sus clientes con el propósito de cubrir las siguientes necesidades generales:

- Se requiere de una aplicación que será una solución integral en materia de registros, modificaciones y consultas de información para el control y administración de las cuentas.
- Un sistema cliente-servidor, con una base de datos para almacenar toda la información.
- Contemple pantallas de captura gráfica tipo Windows para cada uno de los casos que se manejan en las áreas involucradas.
- Se pretende con este sistema que la captura sea rápida y eficiente, con validación para evitar la duplicidad de datos.
- El sistema permitirá crear, modificar y eliminar datos de clientes, cuentas, empleados y movimientos.





- Que contemple una seguridad para todas las áreas involucradas, estableciendo la autenticación de usuarios y nivel de permisos en el sistema con contraseñas de acceso.
- Se podrán obtener e imprimir los reportes y consultas realizadas, lo que nos permitirá la fácil visualización de la información contenida en la base de datos.
- Toda la información tendrá que estar disponible, actualizada y vigente para todos los registros de clientes, cuentas, movimientos y empleados.
- Eliminación de la redundancia de datos.
- Eliminar la recaptura de datos entre las diferentes áreas involucradas.
- Que la información se encuentre en un punto central y pueda ser accesible cuando se requiera desde cualquier área con las respectivas medidas de seguridad.
- Un sistema amigable y de fácil acceso para los usuarios. Crear una interfaz fácil de utilizar para cualquier tipo de persona.
- Que realicé la validación a nivel campo para evitar errores de captura.
- Se adecue a la plataforma tecnológica con la que cuenta actualmente la Sociedad de Ahorro debido a que actualmente el cliente posee las licencias de Microsoft Windows, Office 2007 y Visual Basic 6.0, por lo cual deberá operar con:
  - Ambiente gráfico en Windows de Microsoft
  - Almacenamiento de datos en Microsoft Access
  - Desarrollo del Front-end en Visual Basic 6.0

### 3.3.2 Requerimientos Particulares

El cliente contempla los siguientes requerimientos específicos para el Sistema de Control y Administración de cuentas:

---

---



- El sistema deberá contar con una interfaz de usuario que posea una ventana principal, con una barra de menús que indique todas las funcionalidades del sistema y el acceso a sus principales procesos, que son:

### **Mantenimiento**

- Hacer un nuevo registro.
- Modificar un registro previamente realizado.
- Borrar un registro previamente hecho ó modificado.
- Buscar algún registro dentro de la base de datos.
- Guardar los registros nuevos ó modificados.

### **Registrar**

- Realizar un nuevo movimiento en las cuentas.
- Registrar movimientos de retiro o depósito en las cuentas.
- Registrar fecha, hora y código de empleado que realiza el movimiento.
- Consultar y modificar las fechas y horarios en que se realizan los movimientos.

### **Consultas**

- Realizar consultas por cliente, por empleado, por cuenta o por movimientos realizados.
- Imprimir los reportes de las consultas.
- Exportar los reportes para su análisis.



## Datos Estadísticos

Se generaran los siguientes informes de datos:

- Informe de los movimientos efectuados por mes.
- Resumen de clientes por tipo de cuenta.
- Ingresos generados por tipo de cuenta.
- Resumen de clientes por zona.
- Resumen de ingresos por zona.
- Tipos de cuenta generados por mes.

## Búsqueda

Se podrá realizar la búsqueda de algún registro dentro de la base de datos:

- Búsqueda por zona.
  - Búsqueda por tipo de cuenta.
  - Búsqueda por operación bancaria.
  - Búsqueda por tipo de moneda.
- 
- Contará con un menú de utilidades a fin de facilitar el rápido acceso a herramientas administrativas como calculadora y block de notas para agilizar las funciones de los empleados.
  - Permitirá a los usuarios autenticarse para poder tener acceso al sistema y hacer uso de este, dependiendo de los privilegios de acceso del usuario



- Tener una base de datos actualizada de los clientes de Sociedad de Ahorro que incluya número de cliente, nombre completo, RFC, dirección, teléfono y delegación.
- Llevar un control de las cuentas registradas incluyendo número de cuenta, nombre del cliente, número del empleado que efectuó el movimiento, tipo de moneda (pesos, dólares, euros), tipo de cuenta (ahorro, plazos y monedero electrónico).
- Tener un control de los empleados de la empresa, especificando para cada uno un nivel de acceso y control con contraseñas, para modificar los registros indicando número de empleado, nombre, cargo, dirección, teléfono, sexo, delegación.
- Control de los movimientos que se realizan en las cuentas incluyendo número de cuenta, tipo de movimiento (depósito, retiro), fecha y hora.



### **3.4 IDENTIFICAR LOS POSIBLES MÓDULOS DE LA APLICACIÓN.**

Una de las tareas al realizar un programa o aplicación para una empresa, es identificar las áreas que son punto clave para desarrollar el sistema. Estas áreas o también llamados módulos, pueden ser ejes importantes para que nuestro programa sea funcional.

#### **3.4.1 Definición de Módulo.**

En programación, un módulo es una parte de un programa de ordenador. De las varias tareas que debe realizar un programa para cumplir con su función u objetivos, un módulo realizará una de dichas tareas (o quizá varias en algún caso).

En un caso general, un módulo recibirá como entrada la salida que haya proporcionado un módulo anterior o los datos de entrada al sistema (programa) si se trata del módulo inicial de éste y proporcionará una salida que será utilizada como entrada de un módulo posterior o que será la salida final del sistema (programa) si se tratase del módulo final.

Particularmente, en el caso de la programación, los módulos suelen estar organizados jerárquicamente en niveles, de forma que hay un módulo superior que realiza las llamadas oportunas a los módulos del nivel inferior.



Cuando un módulo es llamado, recibe como entrada los datos proporcionados por el módulo de nivel superior que ha hecho la llamada, realiza su tarea, a su vez este módulo puede llamar a otro u otros módulos de nivel inferior si fuera necesario; cuando finaliza su tarea, devuelve la salida pertinente al módulo superior que lo llamó inicialmente, y es este módulo superior el que continúa con la ejecución del programa.

### 3.4.2 Características de un módulo

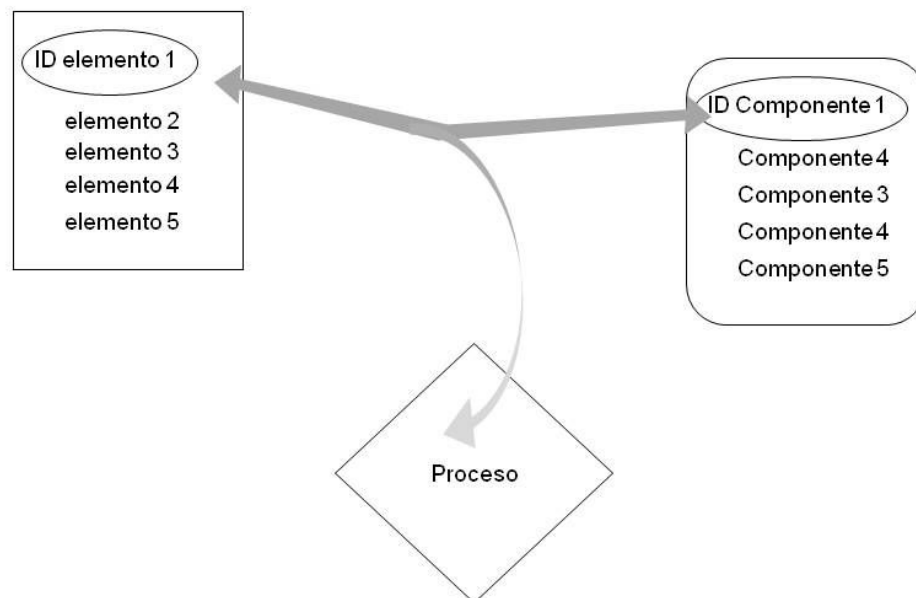
Cada uno de los módulos de un programa idealmente debería cumplir las siguientes características:

- Tamaño pequeño.- Facilita aislar el impacto que pueda tener la realización de un cambio en el programa, bien para corregir un error, bien por rediseño del algoritmo correspondiente.
- Independencia modular.- Cuanto más independientes son los módulos entre sí, más fácilmente se trabajará con ellos, esto implica que para desarrollar un módulo no es necesario conocer detalles internos de otros módulos
- Como consecuencia de la independencia modular, un módulo deberá cumplir con las características de caja negra, es decir con la abstracción, la cual consiste en aislar un elemento de su contexto o del resto de los elementos que lo acompañan y también debe estar encapsulado u oculto, es decir, de los datos miembro de un objeto de manera que sólo se puede cambiar mediante las operaciones definidas para ese objeto.

### 3.4.3 Identificación de módulos.

#### Módulo ID

El **ID (identificador)** es parte primordial en todo elemento; en la figura 3.4.3.1 se ejemplifica la búsqueda de dicho componente, el id, estará presente en la mayoría de la aplicación, se le pueden asignar variables u otro tipo de datos; de igual forma para poder relacionarlo con otros objetos el id, podrá hacer esta conexión y que ambos elementos trabajen entre sí.



**Figura 3.4.3.1 Diagrama de interacción con el Módulo ID.**



## Módulo Clientes

Este módulo es muy importante en nuestro sistema ya que es primordial para operar la información concerniente a los clientes de la Sociedad de Ahorro; los datos del cliente que se desean obtener son:

- Código de Cliente
- Nombre
- Dirección
- Teléfono
- Sexo
- Delegación

## Módulo de Cuenta

El sistema va a ser capaz de registrar movimientos de cuentas financieras, tales como:

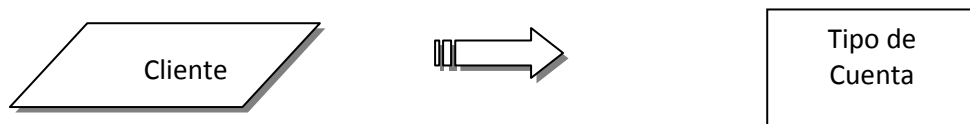
- Ahorro
- Cuentas de ahorro a plazos
- Monedero electrónico

En la figura 3.4.3.2 vemos el proceso de registro de un nueva cuenta cliente y automáticamente se le asigna el tipo de cuenta requerido; este módulo va relacionado con el Módulo Cliente, ya que una nueva cuenta de ahorro se asigna a un cliente.

---

---





**Figura 3.4.3.2. Asignación de Tipo de Cuenta al Módulo Cliente**

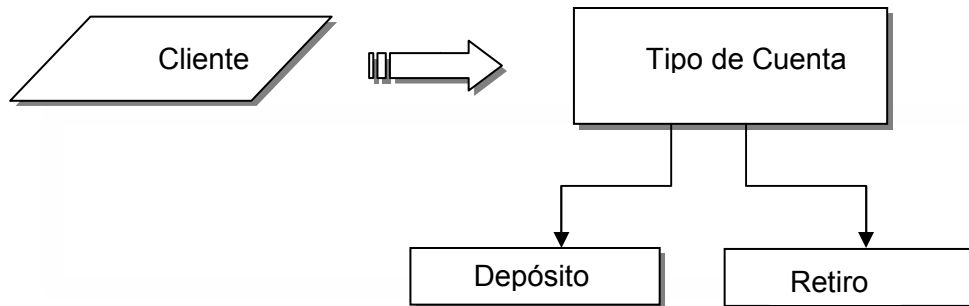
De esta forma podemos ver de una manera sencilla como poco a poco los módulos se van a ir interconectando entre sí, para finalizar creando todo un sistema.

Una vez obtenidos los datos del cliente y el tipo de cuenta que desea operar, se le asigna un consecutivo o folio, que será su identificador para futuras transacciones que esté desee realizar.

### **Módulo de Operaciones**

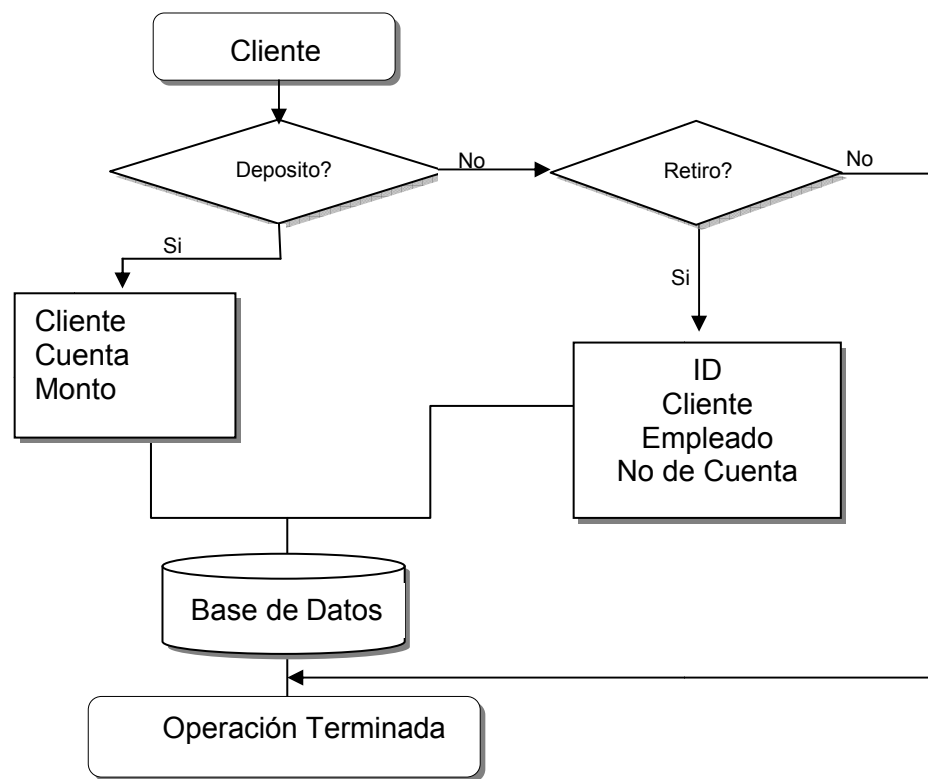
Éste es un módulo permite al cajero registrar el depósito ó retiro de efectivo en las cuentas operadas por la Sociedad de Ahorro. Actualiza los registros, modificando el monto en la cuenta y permite la impresión de recibo para el cliente y una copia para la institución.

En la siguiente figura 3.4.3.3 podemos ver cómo se van enlazando los módulos para poder crear un proceso.



**Figura 3.4.3.3 Formación de módulos del proceso de Alta o Baja de un registro**

En la figura 3.4.3.4 se muestra el diagrama de flujo a partir del cual se registra una operación en el sistema, se encuentran incluidos los módulos, base de datos y procesos.



**Figura 3.4.3.4. Diagrama de Flujo en el registro de una operación financiera**



Dentro de éste diagrama se presenta de manera general el proceso de registro de una operación, englobando los demás datos y que son parte de los módulos que también se han incluido; todo se almacena en la base de datos y finalmente termina la operación.

### **Módulo de Empleado**

De acuerdo a la forma de operar en la empresa “Sociedad de Ahorro”, a todas las nuevas cuentas que se generen, se les asociará el empleado generó ésta, para poder contar con una mejor administración; el módulo empleado debe registrar los siguientes datos:

- Código de Empleado.
- Nombre.
- Cargo.
- Dirección.
- Teléfono.
- Sexo
- Delegación

### **Módulo Cargo del Empleado**

Este es un submódulo de empleado, este módulo es para identificar qué tipo de empleado es, ya que como en toda organización, existen diferentes áreas, y diferentes cargos que tiene la empresa, estos cargos son:



- Cajero
- Gerente
- Subgerente
- Administrador

Es el que finalmente nos hará el registro completo de una nueva solicitud, englobando e interactuando con módulos y con la base de datos.

### **Módulo Listados**

Este módulo es de gran importancia para la empresa, ya que a través de él se podrá conocer el historial y todo lo cargado en la base de datos, este módulo se internará directamente en todas las tablas de la base de datos y arrojará los resultados de búsquedas tales como: clientes, empleados, cuentas y movimientos, ofreciendo la opción a crear reportes y poder visualizar de una mejor forma los status de los clientes.

### **Módulo Datos Estadísticos**

De igual forma que el anterior módulo, pero este con la particularidad de ofrecer reportes por estadísticas, es decir se podrán crear reportes mucho más específicos y más detallados de acuerdo al rubro que se quiera saber, entre ellos se encuentran los siguientes:

---

---



- Resumen de movimientos.
- Resumen por tipo de cuenta.
- Resumen por Delegaciones.
- Estadísticas de cuentas por meses.

### **Módulo de Consulta Avanzada**

Éste módulo nos llevará a través de la aplicación y nos ofrecerá una ventana para buscar datos, podremos buscar datos que fueron almacenados con antelación en el sistema, de una manera mucho más específica, puede ser de acuerdo al apellido de una persona, al nombre, al tipo de cuenta, a la fecha, etc.

### **Módulo de Utilidades**

En este módulo contamos con accesos de apoyo a los empleados, tales como Block de notas, Word, Excel, calculadora y la utilería de reparación de la base de datos a la que solo accede el administrador de la aplicación.

### **Módulo de Ayuda**

En esta opción encontramos la utilería de apoyo para el uso del sistema. Es decir aquí podemos llamar a la guía de usuario del sistema.

## 3.5 JUSTIFICACIÓN DE LA METODOLOGÍA Y DEL SOFTWARE A UTILIZAR.

### 3.5.1 Justificación de la metodología a utilizar.

Una vez analizado el problema y habiendo estudiado las herramientas disponibles, se comienza a estructurar una posible solución, empezando por la metodología a utilizar. Un plan de trabajo tiene el objetivo de dividir, organizar y distribuir tareas específicas de un determinado proyecto con el fin de optimizar los tiempos y recursos que se utilizarán en su desarrollo.

La metodología a utilizar para el desarrollo del Sistema de Administración y Control de Cuentas de Ahorro, es la Metodología de Ciclo de Vida Clásico. Véase Figura 3.5.1.



**Figura 3.5.1 Esquema del ciclo de vida clásico**



Se decidió utilizar el método de vida clásico debido a que se adapta a nuestras necesidades además de que cumple todos los puntos necesarios para el desarrollo de nuestro sistema, en virtud de que consta de las siguientes actividades:

- Investigación preliminar. Cuando se formula la solicitud comienza la primera actividad del sistema. Esta actividad consta de:
    - Aclaración de la solicitud. La solicitud de proyecto debe examinarse para determinar con precisión lo que el solicitante desea.
    - Estudio de factibilidad. En la investigación preliminar un punto importante es determinar que el sistema solicitado sea factible.
  - Determinación de los requisitos del sistema. De común acuerdo con el cliente, se busca llegar a un acuerdo en cómo se resolverá dicho problema. Una vez reunidos los detalles, se estudiarán los datos sobre requerimientos con la finalidad de identificar las características que debe tener el nuevo sistema.
  - Diseño del sistema (diseño lógico). El diseño de un sistema de información responde a la forma en la que el sistema cumplirá con los requerimientos identificados durante la fase de análisis.
  - Se puede realizar un esquema del formato o pantalla que se espera que aparezca cuando el sistema este terminado, se realiza en papel o en la pantalla de una terminal utilizando algunas de las herramientas automatizadas disponibles para el desarrollo de sistemas.
  - Desarrollo de software (diseño del código). Se identifica el software a utilizar y se inicia la programación. La elección depende del costo de cada alternativa, del tiempo disponible para la programación del software y de la disponibilidad de los programadores.
  - Prueba de sistemas. Durante esta fase, el sistema se emplea de manera experimental para asegurarse de que sea consistente y el software no tenga fallas.
- 
-



- Implementación y evaluación. La implementación es el proceso de verificar e instalar la aplicación, así como el entrenamiento a los usuarios y la construcción de todos los archivos de datos necesarios para utilizarla, procurando que el uso inicial del sistema se encuentre libre de problemas, sobre todo que el cliente informe si cumple con la resolución del problema.

Como observamos, la Metodología del Ciclo de Vida Clásico sigue un ciclo de vida en cascada, con etapas bien definidas, lo cual nos indica que para empezar con la siguiente etapa hay que terminar con la actual. Nos sirve para conocer si el proyecto es viable, recoge los requisitos, define la etapa de diseño construcción e implementación del sistema. Por lo anterior se considera la más adecuada para el desarrollo de nuestro sistema.

### **3.5.2 Justificación del software a utilizar.**

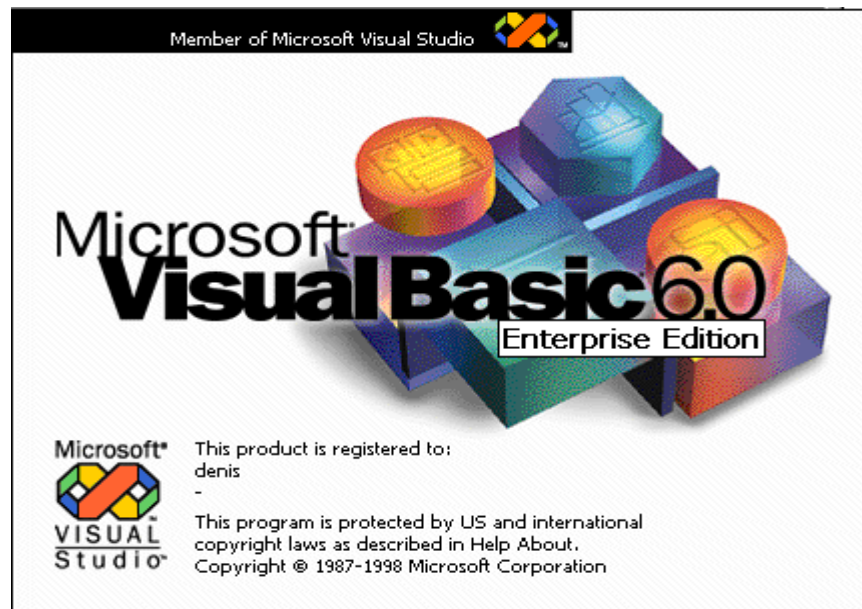
Ya que se ha definido la metodología a utilizar, otro de los aspectos importantes es el software a utilizar.

Para la construcción de un sistema informático la elección de las herramientas a utilizar es una tarea importante. Actualmente el cambio tecnológico, tanto en la parte de hardware como de software, es constante y por lo tanto la oferta es muy amplia. Para llevar a cabo una buena elección se tienen que tomar en cuenta diversos factores tales como el tamaño y tipo de la aplicación deseada, el costo-beneficio, el número de usuarios, el equipo disponible, el tiempo de entrega del sistema y algunos factores adicionales.

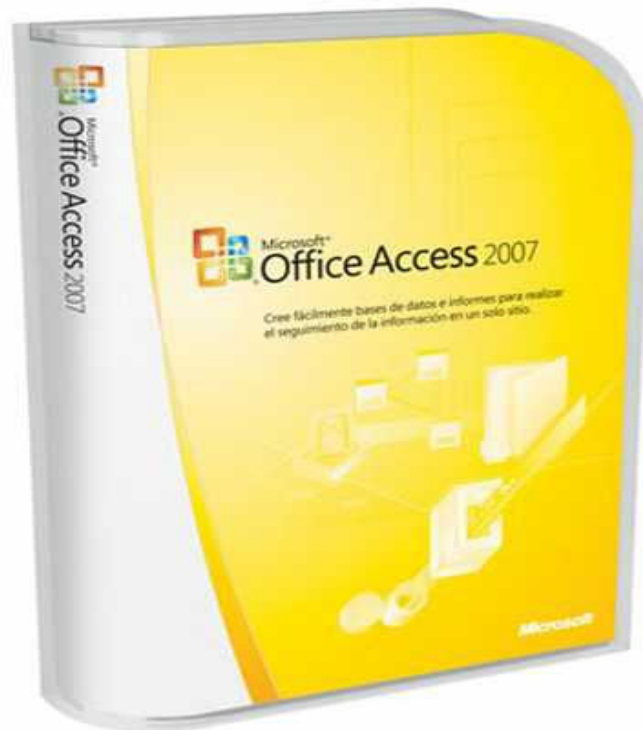




Para el desarrollo del Sistema de Administración y Control de Cuentas de Ahorro se utilizará Microsoft Access 2007 para el desarrollo del back-end y Visual Basic 6.0 para el front-end. Véase Figura 3.5.2 y Figura 3.5.3.



**Figura 3.5.2 Visual Basic 6.0**



**Figura 3.5.3 Microsoft Access 2007**

Lo anterior se debe principalmente a solicitud de nuestro cliente, ya que la Sociedad de Ahorro cuenta con los permisos y licencias necesarios para hacer uso de éste software y se desea reducir costos; aunado a lo anterior, observamos que se ajusta a los requerimientos del sistema por cumplir con las características y ventajas descritas en el capítulo 2 para Visual Basic 6.0 y Microsoft Access 2007.

De la misma manera la empresa no cuenta con un área de sistemas bien estructurada, por tal motivo, estas herramientas son fáciles de mantener y actualizar, por tal motivo cualquier persona con conocimientos básicos en desarrollo de software puede llevar esta tarea sin inconvenientes.



## **CAPÍTULO 4**

### **DISEÑO Y CONSTRUCCIÓN DE LA APLICACIÓN**



## 4.1 ARQUITECTURA DE LA APLICACIÓN.

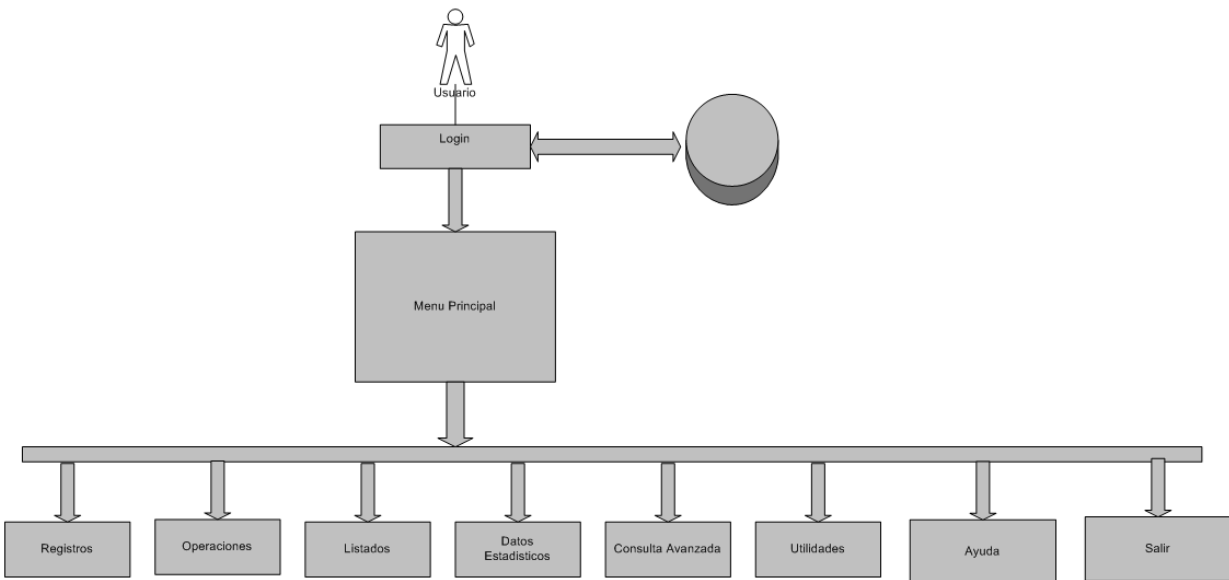
El sistema desarrollado se construyó en forma modular, como dicta la metodología elegida y detallada anteriormente. Durante la etapa de análisis, se plantearon varios posibles módulos de la aplicación. Sin embargo, es durante el diseño y consecuente construcción del programa, que quedan definidos los módulos y submódulos del sistema.

A continuación detallaremos la propuesta final de la aplicación, listando los distintos módulos, para posteriormente detallar cada uno.

- Pantalla de inicio.
- Login (Autenticación).
- Registros.
- Operaciones.
- Listados.
- Datos estadísticos.
- Consulta avanzada.
- Utilidades.
- Ayuda.
- Salida.

### 4.1.1 Pantalla de inicio.

La pantalla de inicio es el punto de partida de la aplicación. Es desde aquí donde se navega hacia cualquiera de los módulos disponibles y acceder a estos. Es ésta la pantalla máster donde se regresa cada vez que finalice alguno de los módulos, y salir de ésta implica salir del sistema. En la Figura 4.1.1.1 podemos observar un diagrama general de la arquitectura propuesta.



**Figura 4.1.1.1 Arquitectura propuesta.**

#### **4.1.2 Login.**

Este bloque de la aplicación es sencillo, sin embargo de vital importancia, ya que evita el uso no autorizado del sistema, así como se asegura de la correcta asignación de privilegios para cada tipo de usuario. Ésta pantalla solo funciona al principio del sistema y hasta que se comprueba la validez de los datos usuario.

#### **4.1.3 Registros**

El módulo de registros, al igual que los siguientes, es accesible mediante la barra de menús del sistema. Este módulo está compuesto de tres submódulos: el acceso al registro de clientes, cuentas y empleados. Estos comparten la misma estructura básica, es decir son capaces de agregar, borrar, editar y modificar un registro existente, y salir.

##### **4.1.3.1 Clientes**

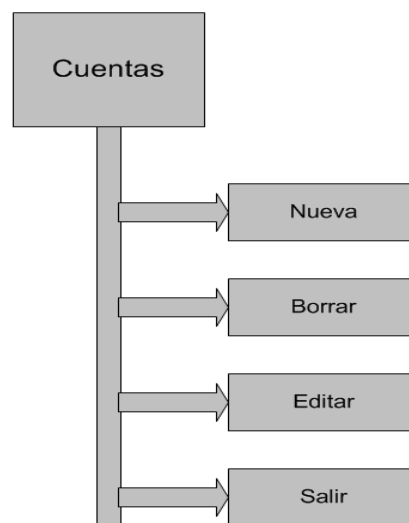
En este submódulo podemos acceder a cuatro funciones: agregar, borrar, editar, cambiar los registros de los clientes y salir. Una de las funciones requeridas del



Este sistema es para manejar los registros de clientes, y este es el módulo necesario para ello, ya que nos permite ejecutar las distintas operaciones sobre los registros, así como navegar entre ellos y realizar búsquedas.

#### 4.1.3.2 Cuentas.

Este submódulo es similar al anterior, de igual forma nos permite las mismas operaciones sobre las cuentas registradas en la base de datos, como se puede apreciar en la figura 4.1.3.2.1; cuenta con las funciones pertinentes y necesarias para el manejo de los productos ofrecidos por la Sociedad de Ahorro.



**Figura 4.1.3.2.1 - Módulo de Registro de Cuentas.**

#### 4.1.3.3 Empleados.

El registro de empleados es el tercer submódulo del módulo de registro y permite el mismo manejo que los dos anteriores, y se adicionaron funciones de búsqueda por cargos, despliegue de los cargos de todos los empleados, mostrando la referencia de sus códigos internos utilizados por la Sociedad de Ahorro.



#### **4.1.4 Operaciones.**

El módulo de operaciones es el encargado de proveer la operación diaria de la Sociedad de Ahorro, es decir, la ejecución de una operación nueva. Este bloque es sencillo, y a la vez el eje central de las operaciones de la Sociedad de Ahorro.

#### **4.1.5 Listados.**

El módulo de listados nos permite ejecutar tres posibles consultas, acerca de la totalidad de clientes, empleados y cuentas. Estos informes son requeridos por las diferentes áreas de la Sociedad de Ahorro, en términos generales, y permiten evaluar las necesidades de capacidad de procesamiento de información, evaluación de personal, de las cuentas existentes, etc.

#### **4.1.6 Datos Estadísticos.**

El uso de reportes es una función importante de un sistema de bases de datos ya que la información almacenada debe poder ser accedida y procesada para que cubra la utilidad que se espera de ésta. Este bloque genera los reportes estadísticos del sistema: históricos de movimientos; parciales mes a mes; un resumen de las operaciones (monto y número de clientes) agrupado por tipo de cuenta y un resumen por delegación.

#### **4.1.7 Consulta Avanzada.**

Éste módulo es de suma utilidad, es la búsqueda principal de datos del sistema. Aquí podemos localizar toda la información del sistema mediante cualquiera de los criterios manejados dentro del mismo, es decir podemos buscar por cliente, número de cuenta, fecha de operación, tipo de cuenta, moneda, o por periodos específicos. Es la búsqueda de información más completa del sistema.



#### **4.1.8 Utilidades.**

Este bloque llama a programas nativos del sistema operativo Windows con el que cuentan las máquinas del cliente donde se ejecuta el sistema (Word, Excel, calculadora y block de notas), con el fin de agilizar las tareas, sin necesidad de salir del sistema. Además cuenta con la utilería de reparación de la base de datos. Esta última es exclusiva para el administrador de la base de datos.

#### **4.1.9 Ayuda.**

Éste módulo permite al usuario realizar consultas sobre la operación del sistema. Se cuenta con explicaciones por función para realizar las tareas necesarias dentro del mismo.

#### **4.1.10 Salir**

El módulo de salida, nos asegura que todas las conexiones a la Base de Datos se cierren de manera adecuada, evitando pérdidas de información y futuros errores.





## 4.2 DIAGRAMACIÓN.

### 4.2.1 Diagrama de contexto.

Todos los proyectos tienen un campo de acción. El ámbito de un proyecto define a qué aspectos de una empresa, se supone, debe dar su apoyo un sistema o una aplicación; y también el modo de interacción del sistema o la aplicación en fase de modernización, con otros sistemas y con la empresa considerada en su conjunto. La definición del ámbito del proyecto es el primer paso importante de la automatización de procesos.

Un diagrama de contexto define el campo de acción y los límites del sistema. El ámbito de todo proyecto está sujeto siempre a cambios; por tanto, también lo deberá estar el diagrama de contexto. Entre sus sinónimos se incluyen diagrama de contexto, modelo de contexto y modelo ambiental.

Un diagrama de contexto contiene un único proceso. Los agentes externos e internos se sitúan en el perímetro del dibujo. Ha dicho perímetro se añaden también los datos externos. Los flujos de datos definen las interacciones del sistema con los agentes internos y externos, y con los datos externos.

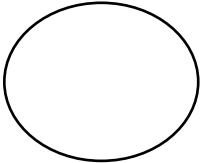


El diagrama de contexto tiene tres entidades importantes con las que debe de interactuar:

- Los demás sistemas con los que interactúa.
- Los usuarios del sistema.
- Los centros de información a los cuales necesita acceder para realizar su función correctamente (Ej.: bases de datos y/o repositorios de información).



Un proceso único representa el sistema externo, definiendo la frontera o el marco del análisis, es decir las interfaces del sistema y el resto del universo.

Los elementos que se emplean para elaborar este diagrama se describen en la Figura 4.2.1.1.

SIMBOLO	SIGNIFICADO
	Un proceso significa que se realizan algunas acciones o grupo de acciones que transformarán o modificarán la información o datos.
	Una entidad es una persona, grupo, departamento o cualquier sistema que recibe u origina información o datos.
	Un flujo de datos muestra que es transferida información desde o hacia un proceso.

**Figura 4.2.1.1 Notación grafica usada en un diagrama de contexto.**

En la figura 4.2.1.2 se muestra el diagrama de contexto para el Sistema de Administración y Control de Cuentas de Ahorro. Se observa que dentro de la burbuja principal está nuestro sistema, con los componentes que interactúan directamente con él. Se tomó en cuenta que tanto otros programas, como usuarios (llámese administrador, empleado, etc.) interactúan con nuestro sistema y con ello se complementan las funciones y operaciones requeridas.

En este diagrama se presenta un panorama general de las diferentes entidades que intervienen en el sistema, así como el flujo de información entre ellos, lo cual significa

que nuestro sistema recibirá datos de las entidades, los procesara y emitirá resultados que involucren a otras entidades o a la misma que generó la información.

Se representan los elementos básicos, donde los usuarios acceden al sistema y es aquí donde se involucran las demás áreas relacionadas en este proceso, del cual se pueden ir desarrollando los eventos del sistema.



**Figura 4.2.1.2 Diagrama de Contexto General del Sistema Nivel 0.**

#### **4.2.2 Diagrama de Procesos.**

Se llama Diagrama de Procesos a la representación gráfica de los pasos que se siguen en toda una secuencia de actividades, dentro de un proceso o un procedimiento, identificándolos mediante símbolos de acuerdo con su naturaleza.


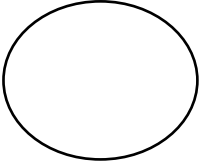



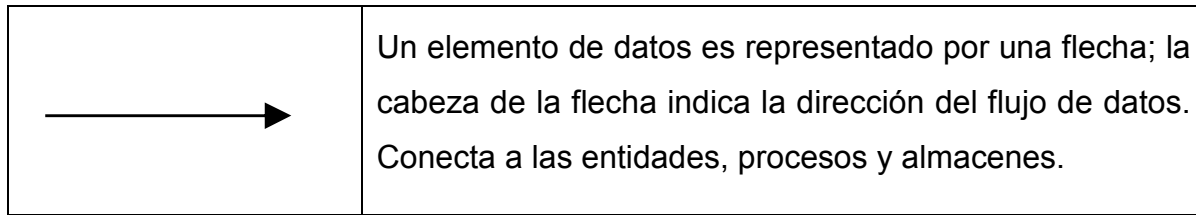
Sirve para fines de planificación y como ayuda para descubrir y eliminar ineficiencias, es conveniente clasificar las acciones que tienen lugar durante un proceso.

Las características que debe cumplir un diagrama de procesos es la siguiente:

- Las operaciones deben mostrarse mediante bloques
- La mayoría de las líneas de flujo se representan con flechas que van en dirección del flujo.
- La dirección del flujo es de izquierda a derecha mientras sea posible.
- Se suministra únicamente información relevante del proceso.
- Si las líneas se cruzan, la línea horizontal es continua y la línea vertical es corta.

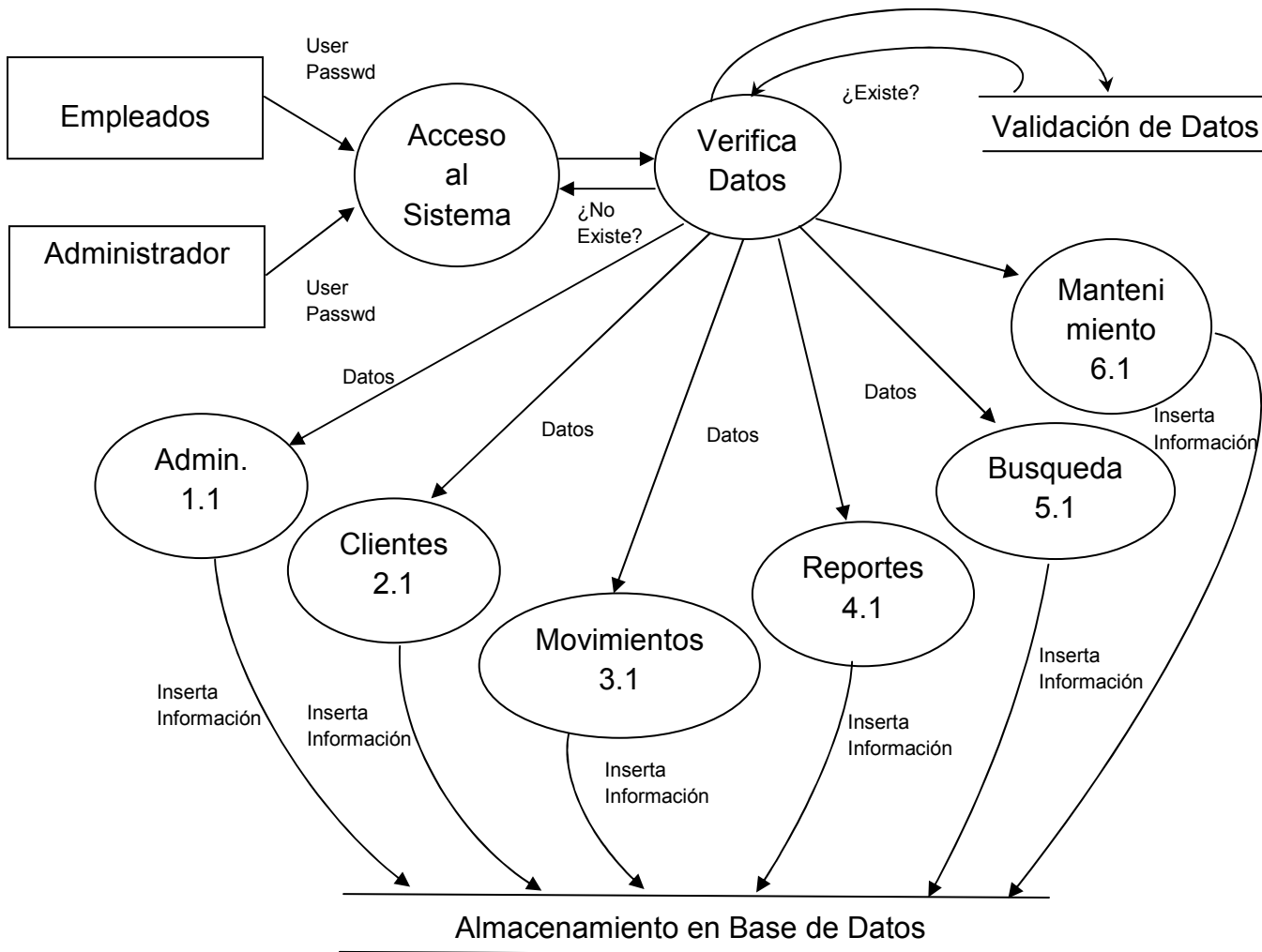
Los elementos que se emplean para elaborar este diagrama se muestran y se describen en la figura 4.2.2.1.

<b>SIMBOLO</b>	<b>SIGNIFICADO</b>
	Una entidad externa es un elemento del sistema, generalmente son personas, organizaciones, o sistemas de cómputo externos que pueden enviar o recibir datos del sistema.
	Un proceso se indica mediante un círculo, los procesos denotan un cambio o transformación de los datos.
	Un almacén de datos es representado por dos líneas paralelas. Se utiliza para almacenar datos.



**Figura 4.2.2.1 Notación grafica usada en un diagrama de procesos.**

En la figura 4.2.2.2 se muestra el Diagrama de Procesos para el Sistema de Administración y Control de Cuentas de Ahorro, en ella van incluidos todos los procesos hasta el primer nivel.



**Figura 4.2.2.2 Diagrama de Procesos, Nivel 1**



En el diagrama anterior podemos constatar la interacción de las entidades externas como lo son los empleados y el administrador; los procesos, que son el acceso al sistema, la validación de datos, la administración, los clientes, el tipo de cuenta, los reportes, la búsqueda de datos y el mantenimiento del sistema; el almacenamiento de la información en la base de datos y el elemento de datos que es simbolizado por las flechas que son las encargadas del transporte de datos.

### 4.2.3 Módulo de Administración

Para el correcto análisis del sistema se deben desglosar a detalle cada una de las burbujas o procesos, el primer módulo de Administración se representa en la figura 4.2.3.1

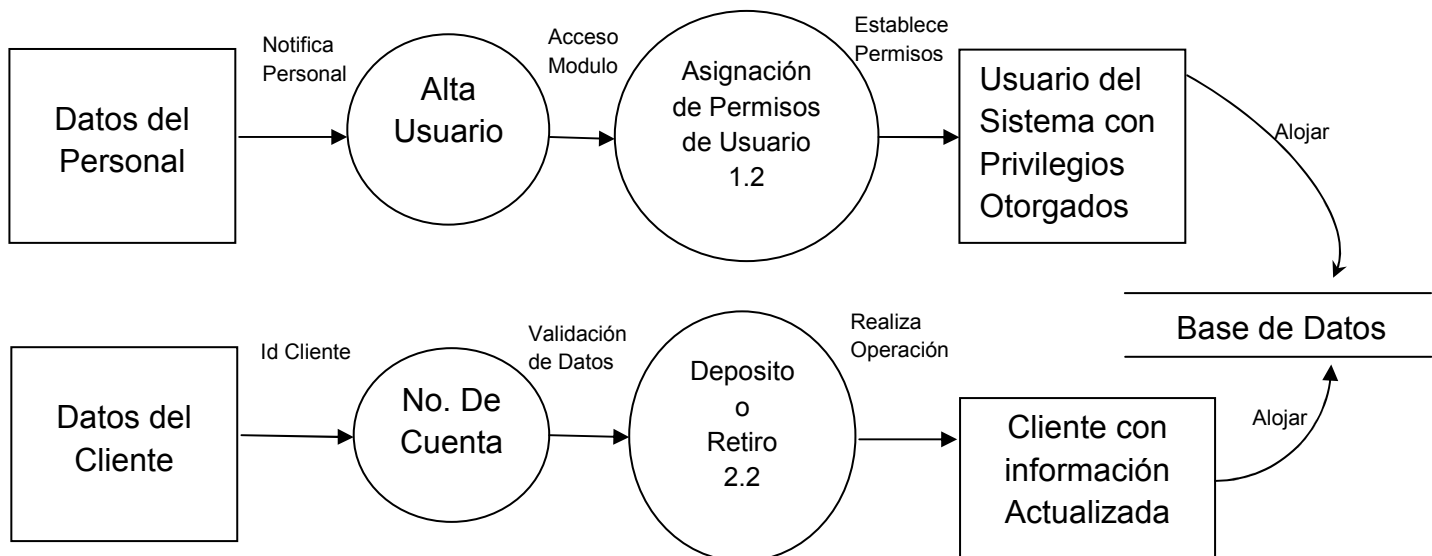
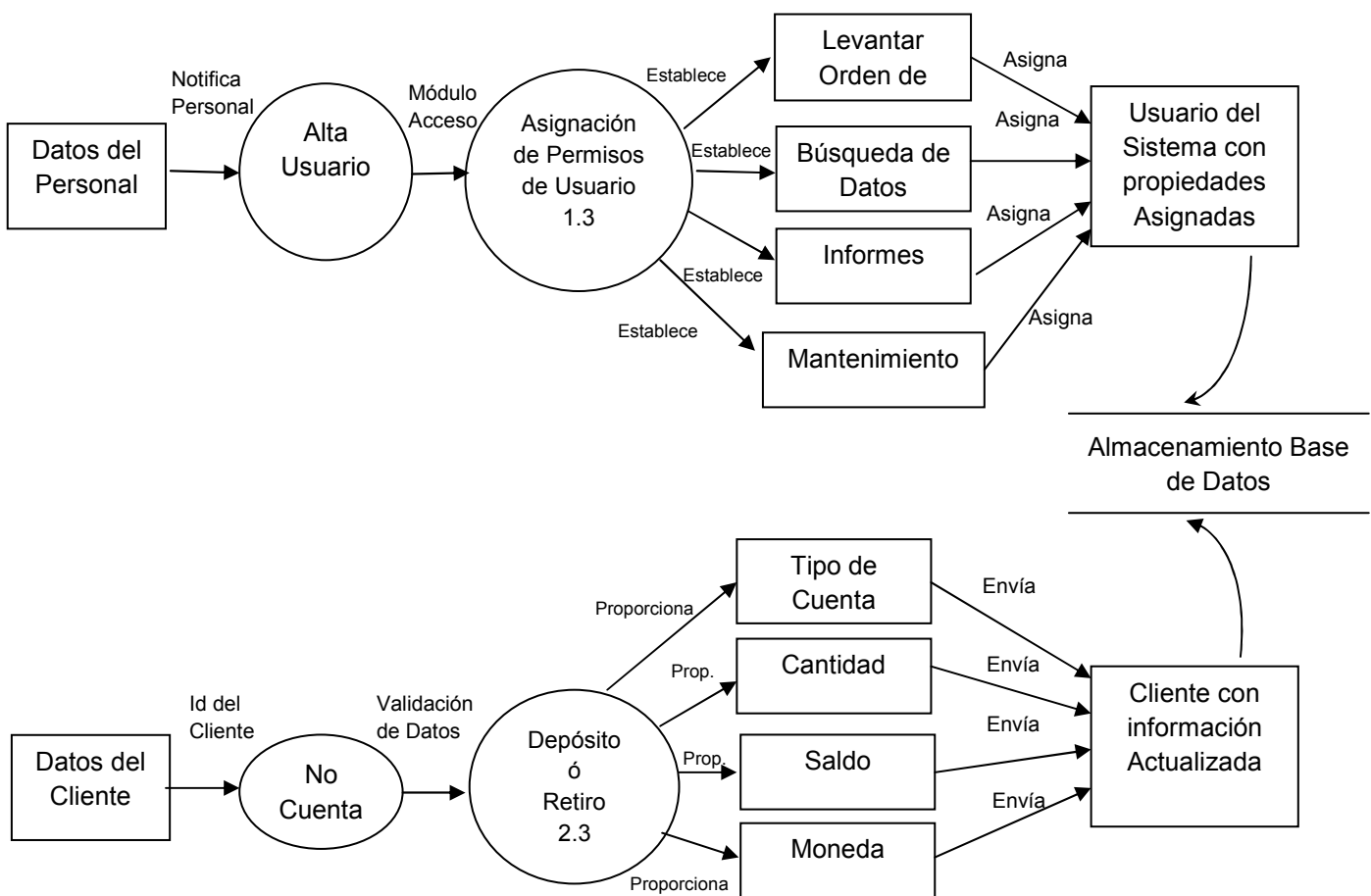


Figura 4.2.3.1 Diagrama de Procesos Nivel 2, Módulo Movimientos.



Como se muestra en la figura anterior, primero se ingresan los datos del personal que hará uso del sistema, con permisos de acceso previamente autorizados, los cuales están almacenados en la base de datos. Un proceso similar se sigue cuando el cliente realiza un depósito, se le solicitan sus datos, se ingresa el número de cuenta, se realiza la operación y posteriormente el sistema hace el cambio y actualiza la base de datos.

En la figura 4.2.3.2 se muestra el diagrama de procesos desarrollado hasta el nivel 3, donde podemos notar más a detalle el movimiento de los procesos o de los datos, y en donde finalmente todo sigue siendo almacenado en base de datos, para que al reiniciar cualquier operación, el sistema mantenga actualizada toda la información que se va ingresando, ya sea para depósito o retiro de un monto de dinero en específico.



**Figura 4.2.3.2 Diagrama de Procesos Nivel 3, Módulo Administración.**

#### 4.2.4 Módulo Reportes

Los reportes son esenciales en cada organización, sirven para visualizar o monitorear el comportamiento de la empresa, a través de ellos observaremos comportamientos que no son visibles durante los procesos normales de trabajo. En las figuras 4.2.8.1 y 4.2.8.2 el diagrama de procesos de segundo y tercer nivel respectivamente.

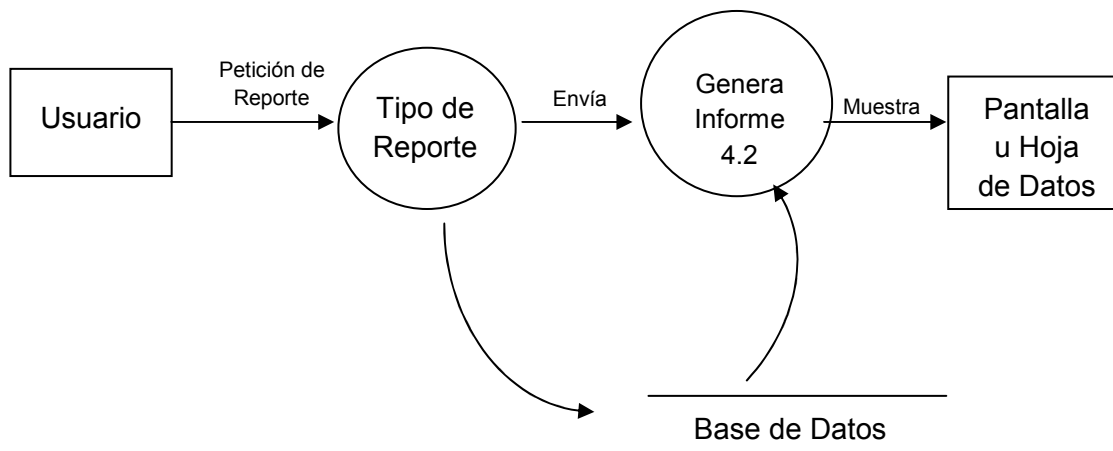


Figura 4.2.8.1 Diagrama de Procesos Nivel 2, Módulo Reportes.

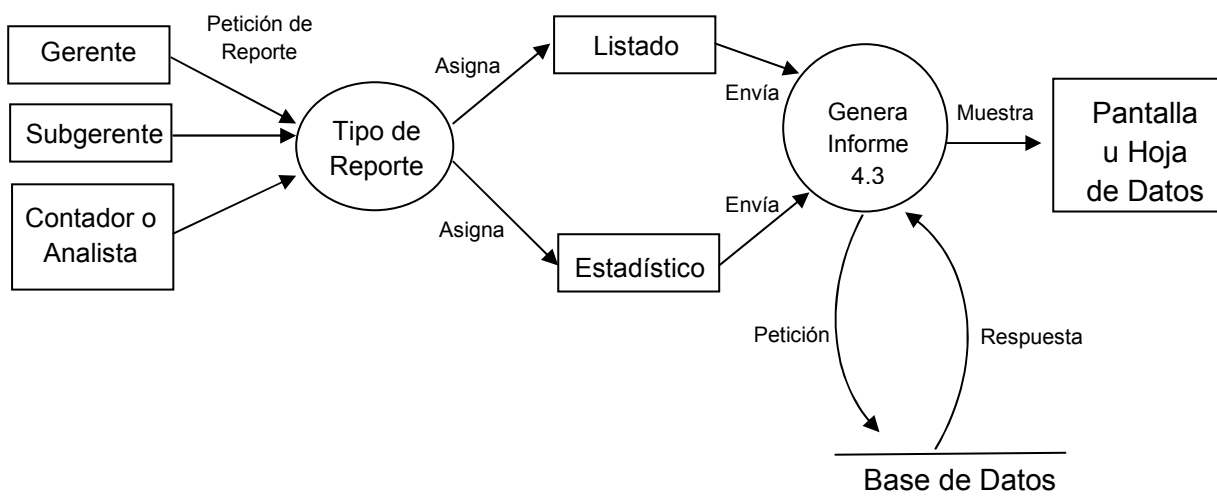


Figura 4.2.8.2 Diagrama de Procesos Nivel 3, Módulo Reportes.



#### 4.2.5 Módulo Búsqueda

Para todo sistema, diccionario electrónico o procesador de documentos es importante un mecanismo de búsqueda capaz encontrar cualquier dato que el usuario requiera. Es por ello que al sistema se le incorporó un buscador de datos; este módulo de búsqueda es capaz de encontrar cualquier información que fue alojada en la base de datos. Su metodología se describe en la figura 4.2.5.1, que es el diagrama de procesos nivel 2 y posteriormente, a detalle, en la figura 4.2.5.2 se muestra el diagrama de procesos de nivel 3.

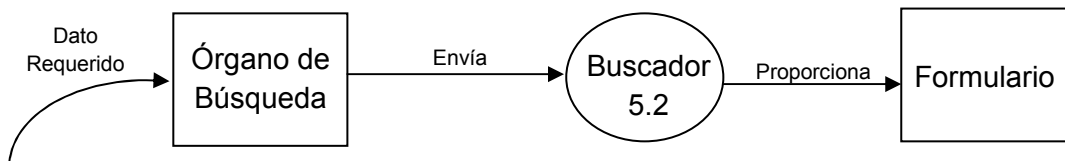


Figura 4.2.5.1 Diagrama de Procesos Nivel 2, Módulo Búsqueda.

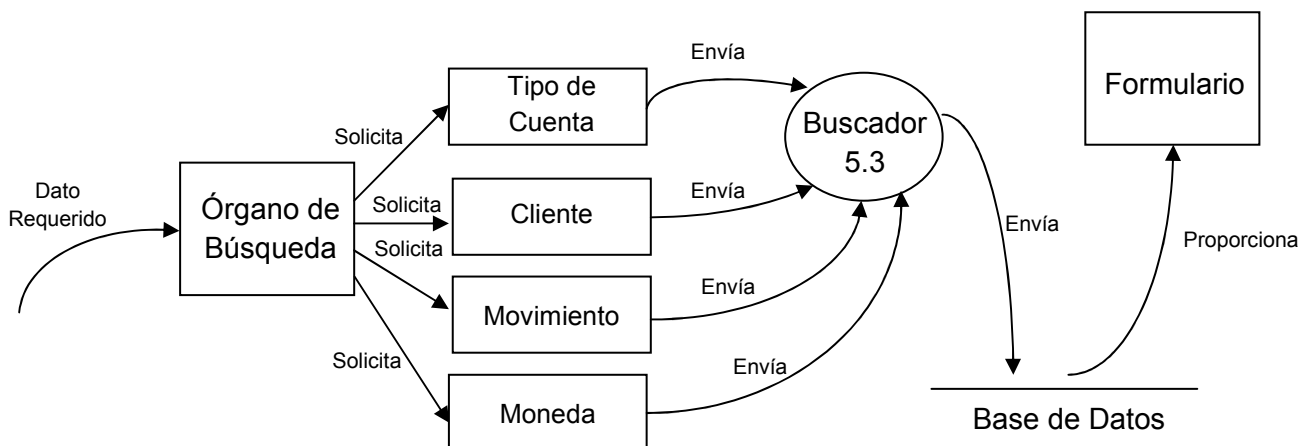


Figura 4.2.5.2 Diagrama de Procesos Nivel 3, Módulo Búsqueda.

### 4.2.6 Módulo Mantenimiento

Este módulo presentado en el diagrama 4.2.6.1 nos muestra el proceso con el cuál se va a reparar la base de datos si se encuentra averiada por mal uso o como manera preventiva, muy útil para evitar futuros daños en la estructura de la base y para desfragmentación y compactación de dicha base, ayuda en la reducción de espacio en disco.

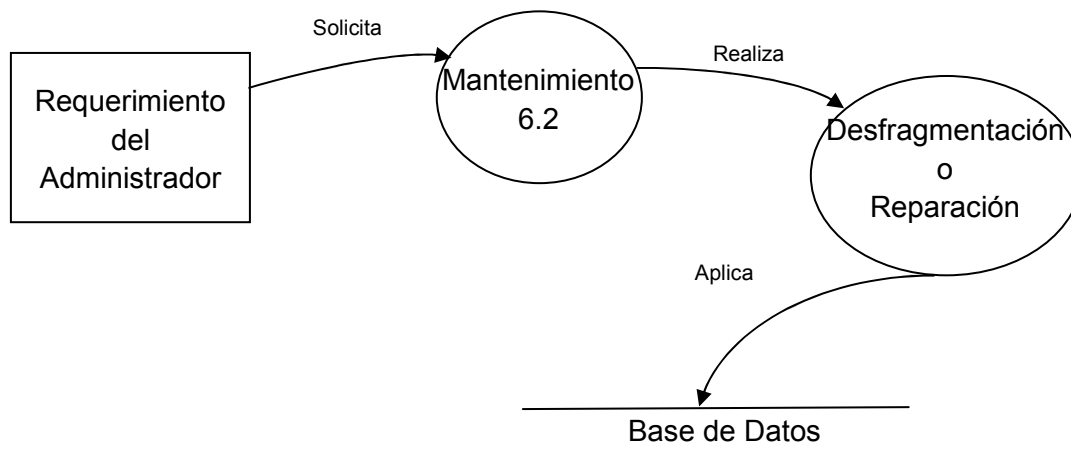


Figura 4.2.6.1 Diagrama de Procesos Nivel 2, Módulo Mantenimiento.

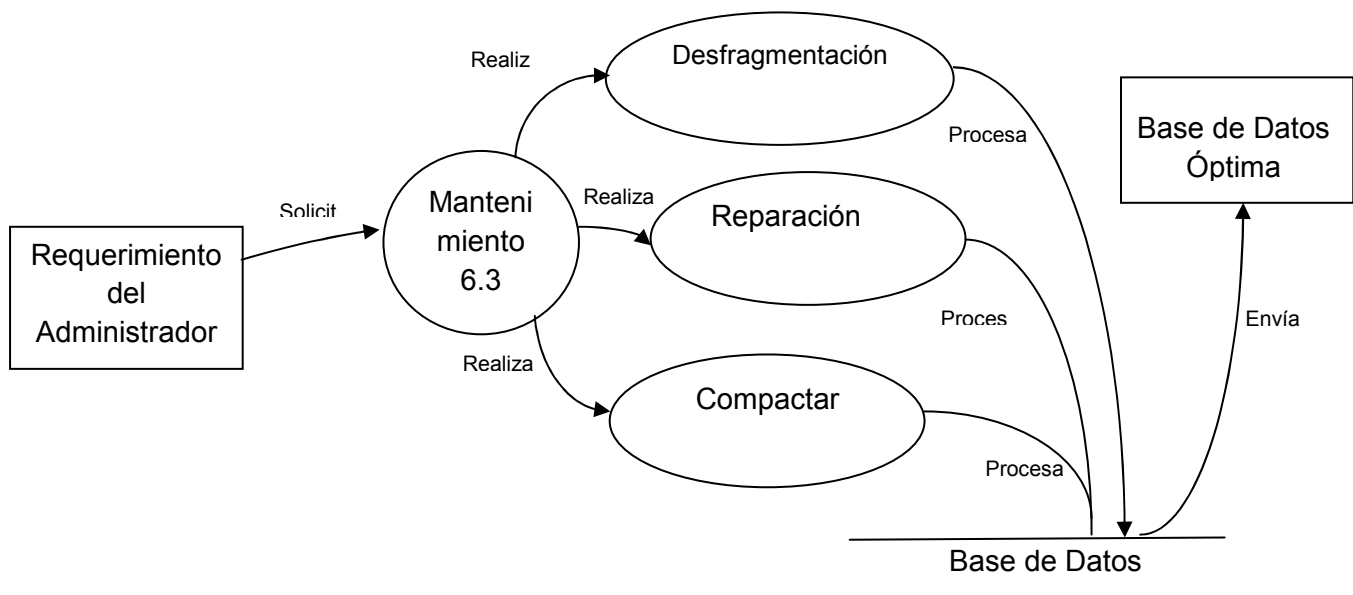


Figura 4.2.6.2 Diagrama de Procesos Nivel 3 Módulo, Mantenimiento.



#### 4.2.7 Modelado de una base de datos

El modelado de datos es una colección de herramientas conceptuales para describir los datos, las relaciones que existen entre ellos, semántica asociada a los datos y restricciones de consistencia. Un modelo es una representación de la realidad que contiene las características generales de algo que se va a realizar. En una base de datos, esta representación se elabora de forma gráfica.

- **Modelos lógicos basados en objetos**

Se usan para describir datos en los niveles conceptual y de visión, es decir, con este modelo representamos los datos de tal forma como nosotros los captamos en el mundo real, tienen una capacidad de estructuración bastante flexible y permiten especificar restricciones de datos explícitamente. Existen diferentes modelos de este tipo, pero el más utilizado por su sencillez y eficiencia es el modelo Entidad-Relación.

- **Modelo Entidad – Relación**

Denominado por sus siglas como: E-R. Este modelo representa a la realidad a través de entidades, que son objetos que existen y que se distinguen de otros por sus características. Las entidades pueden ser de dos tipos:

- **Tangibles:** Son todos aquellos objetos físicos que podemos ver, tocar o sentir.
- **Intangibles:** Todos aquellos eventos u objetos conceptuales que no podemos ver, aún sabiendo que existen.

Las características de las entidades en una base de datos se llaman atributos. A su vez, una entidad se puede asociar o relacionar con más entidades a través de relaciones. La relación entre ambas entidades se establece como operación.

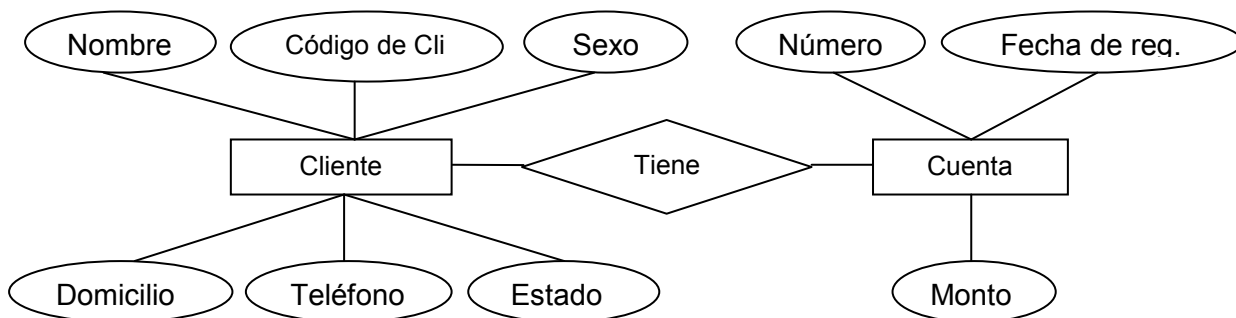
Ahora falta describir como se representa un modelo E-R gráficamente; la representación es sencilla, se emplean símbolos, como se muestra en la Figura 4.2.7.1.

Símbolo	Representa
	ENTIDAD
	RELACIÓN
	ATRIBUTO
	LIGAS

**Figura 4.2.7.1 Símbolos y descripción de un diagrama entidad-relación**

- **Aplicación del modelo E-R en el sistema de administración y control de cuentas de ahorro.**
  - Cliente-Cuenta, E-R 1

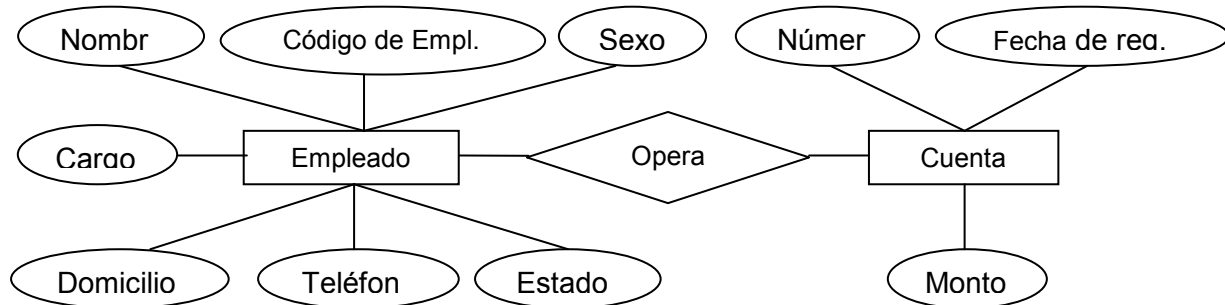
El modelo de la Figura 4.2.7.2, define la relación cliente-cuenta, donde las entidades – con sus atributos- son cliente y cuenta y la relación es: “tiene”.



**Figura 4.2.7.2 Diagrama Entidad-Relación “Cliente-Cuenta”**

- **Empleado-Cuenta, E-R 2**

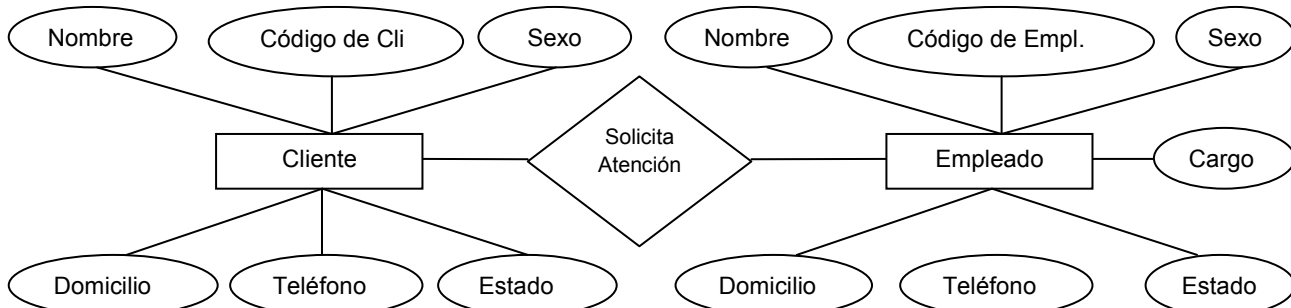
El modelo de la Figura 4.2.7.3, describe la relación empleado-cuenta donde las entidades –con sus atributos- son empleado y cuenta y la relación es “opera”.



**Figura 4.2.7.3 Diagrama Entidad-Relación "Empleado-Cuenta"**

➤ **Cliente-Empleado/Empleado-Cliente, E-R 3**

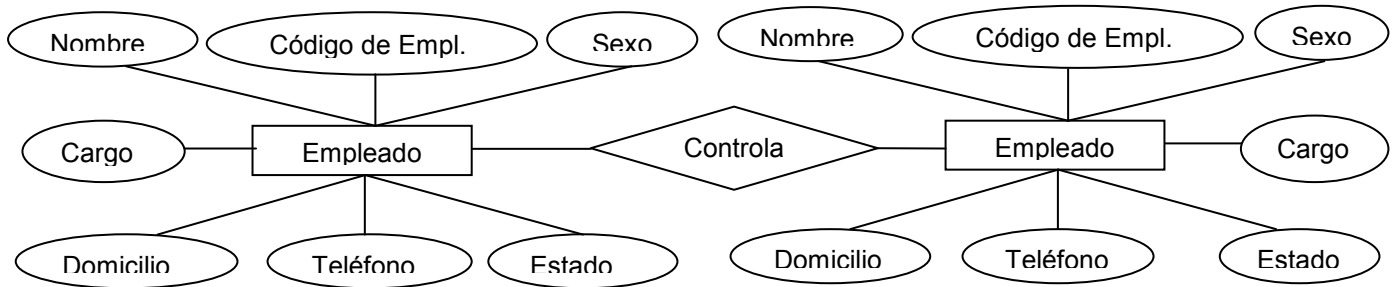
El siguiente diagrama relacional, Figura 4.2.7.4, representa la relación cliente-empleado, donde la relación se establece cuando el cliente recibe atención y el empleado la proporciona.



**Figura 4.2.7.4 Diagrama Entidad-Relación "Cliente-Empleado"**

➤ **Empleado-empleado, E-R 4**

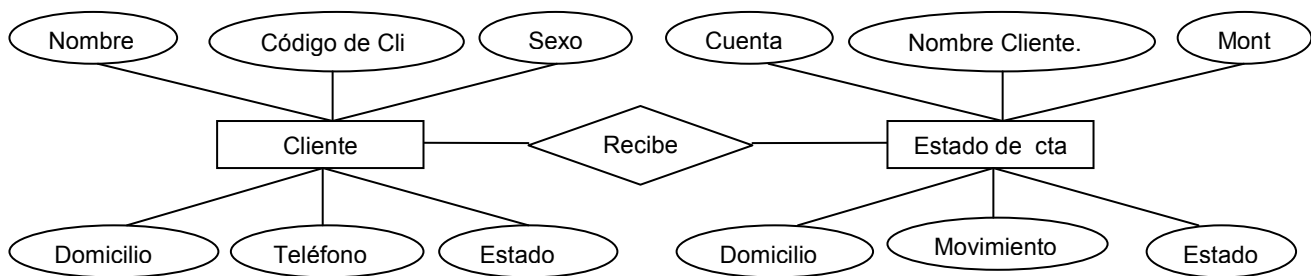
La entidad-relación empleado-empleado se establece cuando un empleado de mayor rango "controla" a otro de menor rango. Esto implica que un gerente puede dar de alta a otro empleado en el sistema. Ver figura 4.2.7.5.



**Figura 4.2.7.5 Diagrama Entidad-Relación "Empleado-Empleado"**

➤ **Cliente-Estado de cuenta, E-R 5:**

La relación cliente-estado de cuenta queda establecida cuando el primero "recibe" el segundo, correspondiente al mes anterior. La Figura 4.2.7.6 describe la relación.



**Figura 4.2.7.6 Diagrama Entidad-Relación "Cliente-Estado de cuenta"**

➤ **Empleado-Estados de cuenta, E-R 6**

La relación "Empleado-Estado de Cuenta" queda establecida cuando el empleado "genera" los estados de cuenta de los clientes. Ver figura 4.2.7.7.

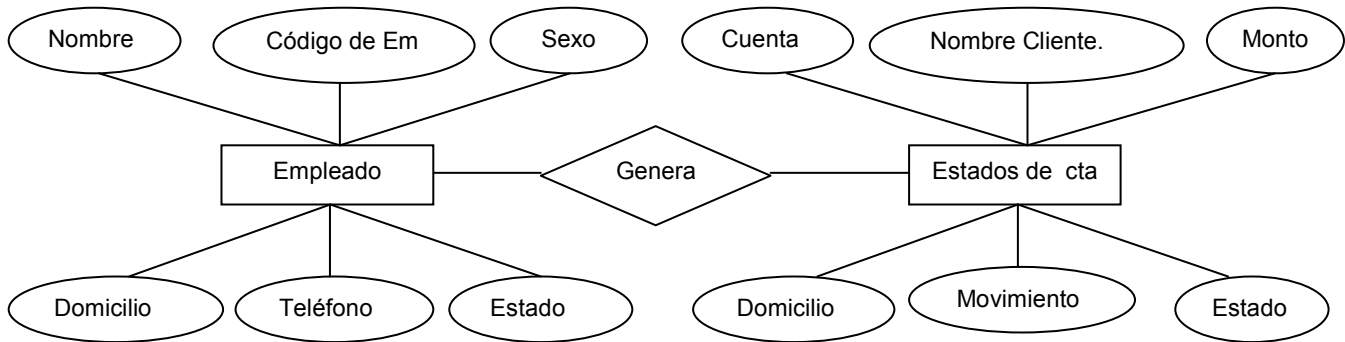


Figura 4.2.7.7 Diagrama Entidad-Relación “Empleado-Estado de cuenta”

➤ Empleado-Información, E-R 7

El empleado y la información están relacionados a través de dos acciones: “generar” y “consultar”. La primera se da cuando por requerimientos de la institución se genera información (estadística, informativa, etc). La segunda se da cuando por necesidad se requiere “consultar” ésta. Los objetivos de la consulta pueden ser para toma de decisiones o verificar alguna situación en particular. Ver figura 4.2.7.8.

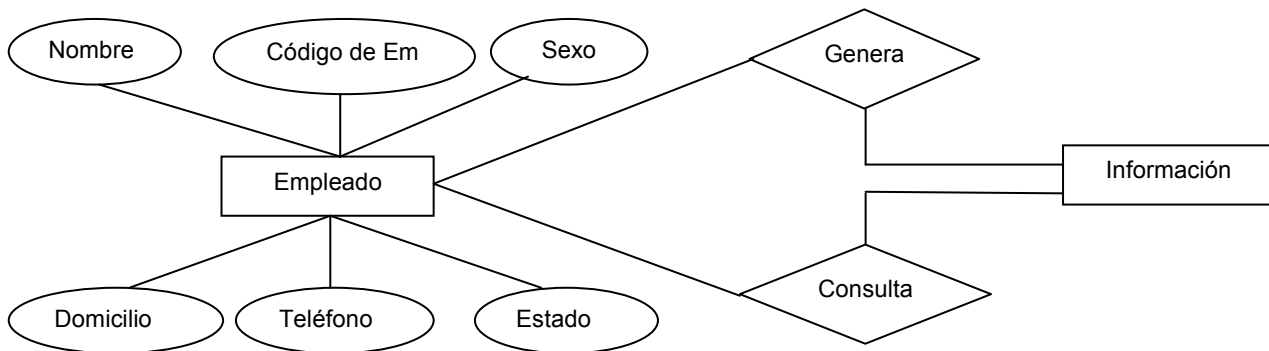
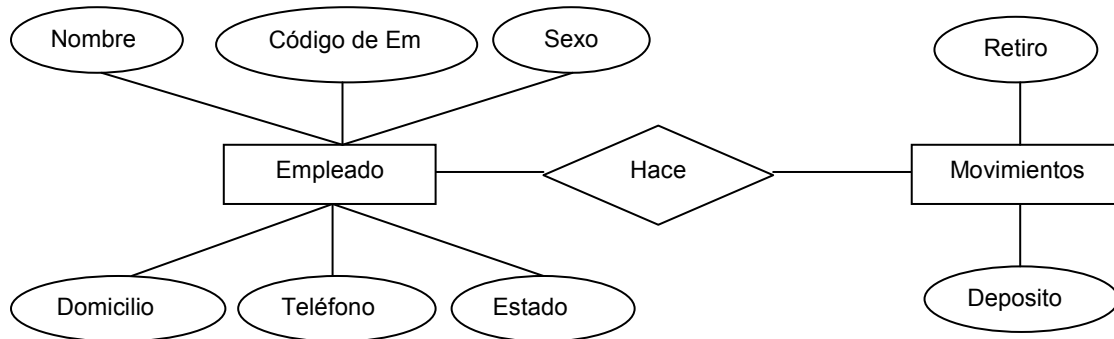


Figura 4.2.7.8 Diagrama Entidad-Relación “Empleado-Información”

➤ Empleado-Movimientos, E-R 8

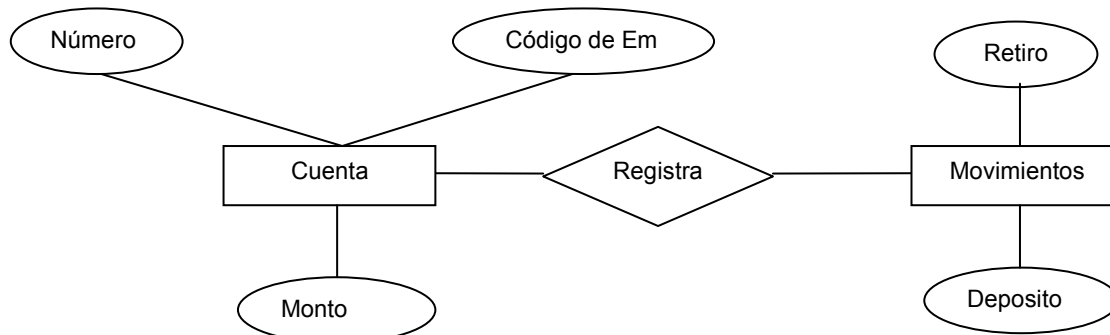
Los movimientos en una cuenta generalmente son realizados por el cajero en turno, pero bien pueden ser hechos por otros empleados de mayor rango. Así la relación está dada por “hace”, como se muestra en la figura 4.2.7.9.



**Figura 4.2.7.9 Diagrama Entidad-Relación “Empleado-Estado de cuenta”**

➤ **Cuenta-Movimientos, E-R 9**

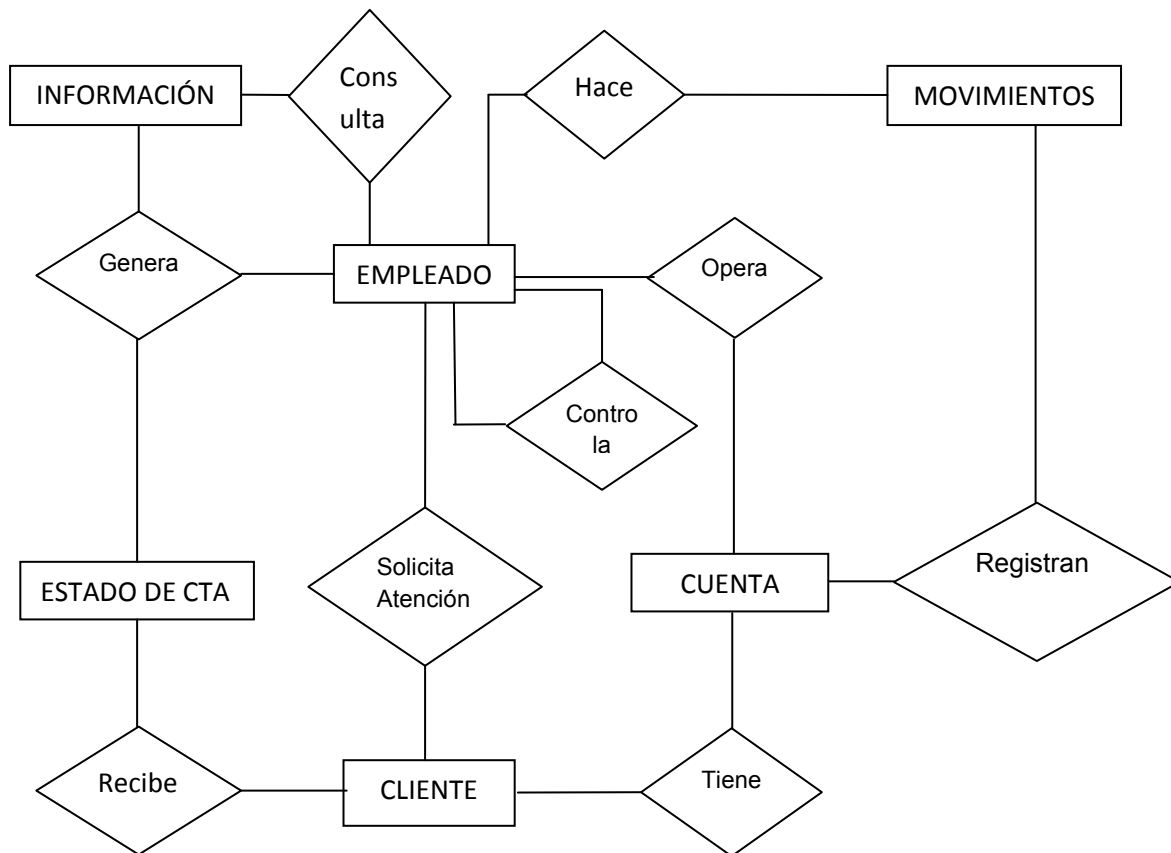
Los movimientos quedan “registrados” en la cuenta, cada vez que el cajero realiza uno. La operación “registra” relaciona a la cuenta con los movimientos, como se muestra en la figura 4.2.7.10.



**Figura 4.2.7.10 Diagrama Entidad-Relación “Cuenta-Movimientos”**



Uniendo los E-R (1-9) generamos el E-R del sistema. Como pudo verse, durante el desarrollo de los diagramas entidad-relación, hay entidades comunes. A partir de dichas entidades se crea el diagrama entidad-relación que describen el diseño de la base de datos y se muestra en la Figura 4.2.7.11.



**Figura 4.2.7.11 Diagrama Entidad-Relación del sistema**

El modelo anterior se ve reflejado en el esquema relacional de la base de datos del sistema en la figura 4.2.7.12. En el pueden observarse, claramente, las ligas que establecen la relaciones entre tablas y que permiten al sistema el manejo de información de forma eficiente. Estas se dan gracias a que dos tablas, al menos, tienen un atributo en común. Existen relaciones entre tablas, que no aparecen descritas en el diagrama anterior, que son las establecidas entre sus atributos y todos sus posibles valores, entre un atributo y el catálogo de opciones que puede tomar dicho atributo.

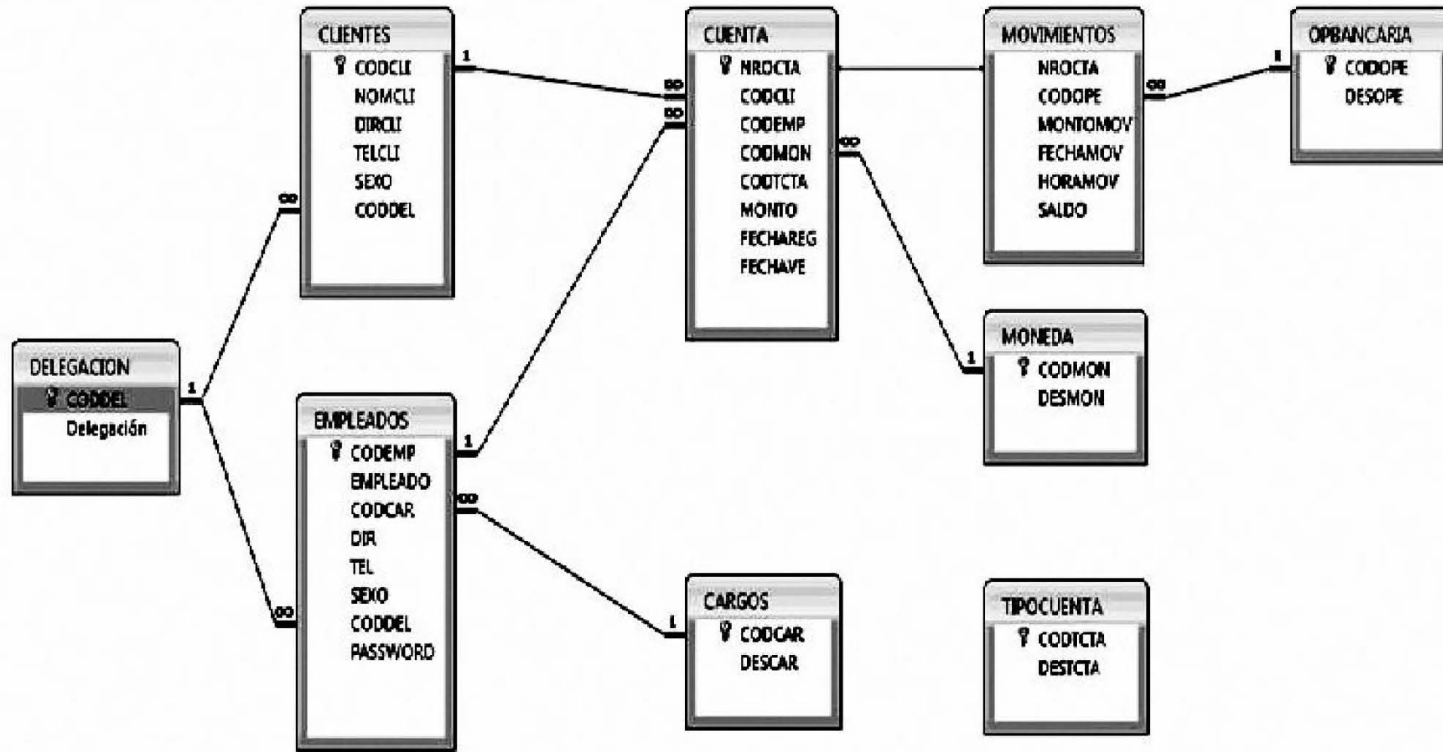


Figura 4.2.7.12 Diagrama Entidad-Relación del sistema de administración y control de cuentas de ahorro



### 4.2.8 Diagramas de Flujo del Sistema de Administración y Control de Cuentas de Ahorro

La conformación general del sistema está compuesta de la siguiente manera, como lo muestra la Figura 4.2.8.1

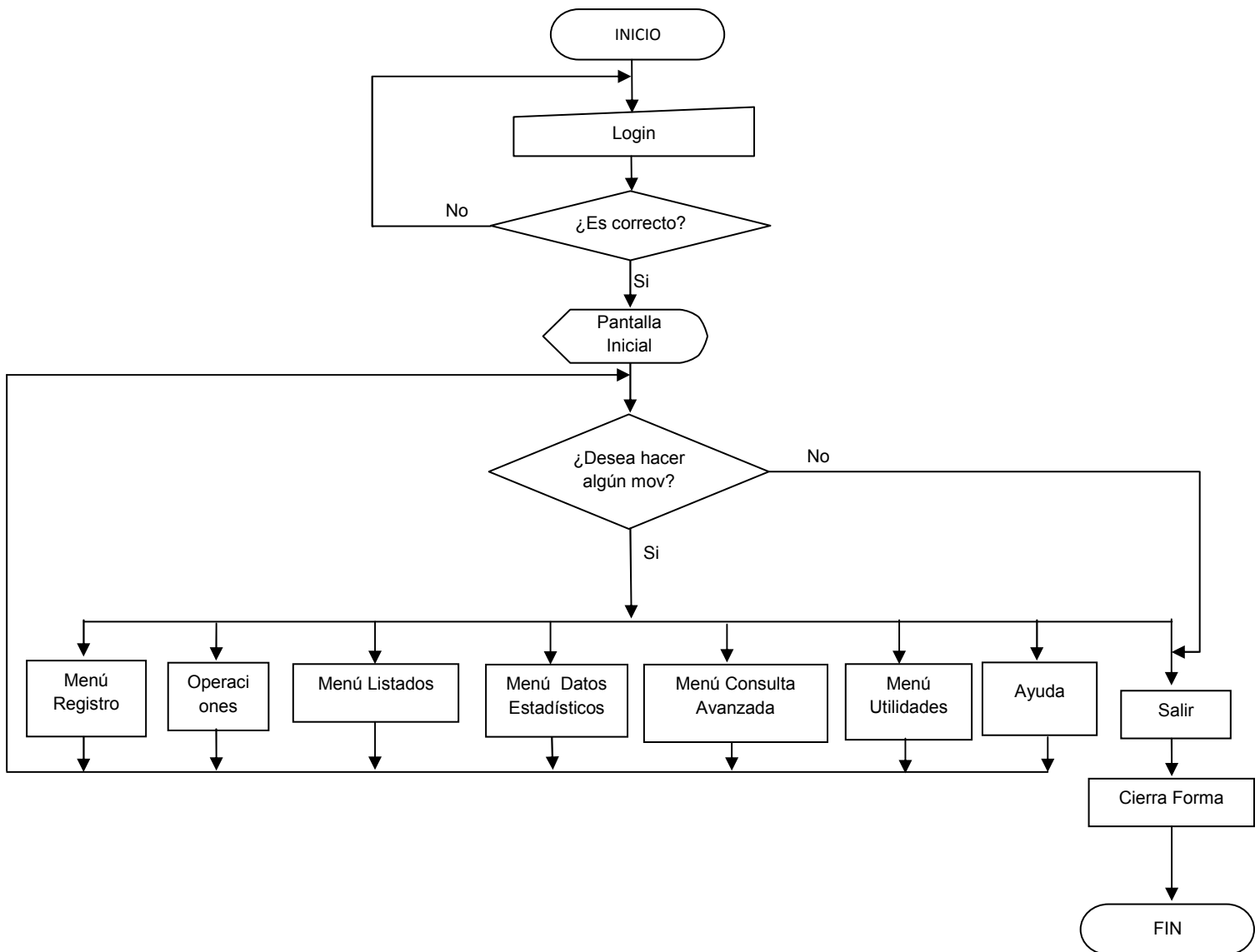


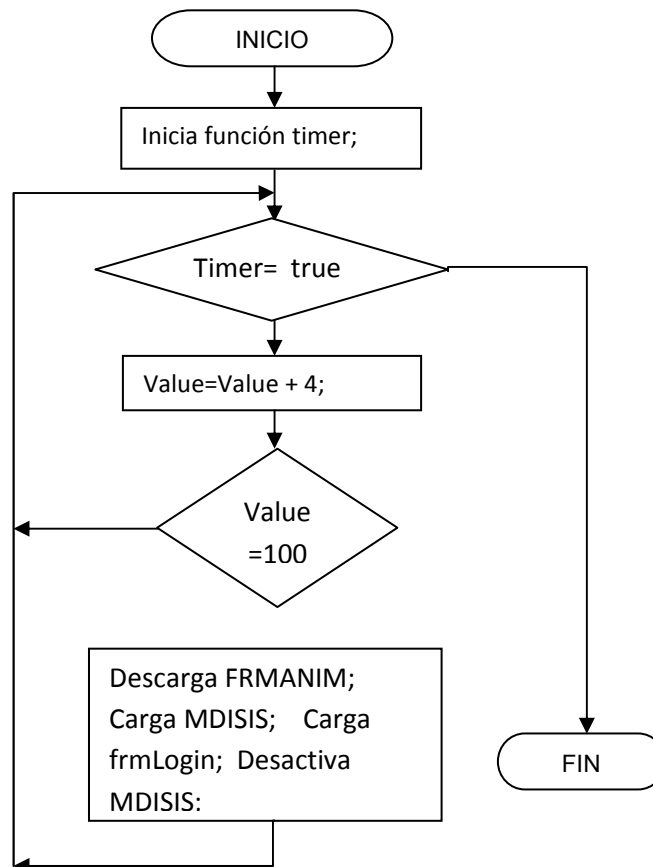
Figura 4.2.8.1 Diagrama de flujo general del sistema de cuentas de ahorro



- **Sistema.-** Inicio del sistema de administración y control de cuentas de ahorro. Abre la base de datos, pantalla de inicio, solicita el login, password y da acceso al menú principal.
- **Registro.-** Este módulo a su vez cuenta con tres submódulos para el control de clientes, cuentas y empleados; cada uno de éstos efectúa las operaciones de alta, bajas y cambios sobre los objetos enunciados.
- **Operaciones.-** En éste módulo se realizan las operaciones de depósito y retiro sobre cuentas de ahorro, monedero y a plazos en pesos, dólares y euros.
- **Listados.-** Este módulo nos permite visualizar los listados completos de los clientes, cuentas y empleados registrados en la base de datos, para posteriormente imprimirlos o exportarlos a formato Excel para su manipulación.
- **Datos Estadísticos.-** El módulo permite extraer información de la bases de datos para uso estadístico, de tal forma que se extrae información de movimientos, movimientos por mes, resumen por tipos de cuenta, por delegación, y cuentas por meses.
- **Búsqueda de Datos.-** La búsqueda de datos se realiza en éste módulo utilizando los siguientes criterios: Cliente, número de cuenta, delegación, moneda, tipo de cuenta, operación bancaria, fecha y periodos de fecha.
- **Utilidades.-** Permite al usuario, auxiliarse con herramientas como calculadora, bloc de notas, Word y Excel.
- **Ayuda.-** Contiene un manual para el uso del sistema.

Inicio del sistema de administración y control de cuentas de ahorro. Abre la base de datos, pantalla de inicio, solicita el login, password y da acceso al menú principal.

- **FRMANIM**



**Figura 4.2.8.2 Diagrama de flujo de FRMANIM**

La figura 4.2.8.2 muestra el diagrama de flujo correspondiente.

- Una vez que inicia, activa la función de tiempo.
- El registro PRBSIS cuenta con un contador Value el cual se incrementa en 4 unidades cada ciclo.
- Si Value es igual a 100, Timer termina, descarga FRMANIM y carga las pantallas de MDISIS y frmLogin. Sino continua activa PRBSIS.
- Desactiva MDISIS.

• frmLogin

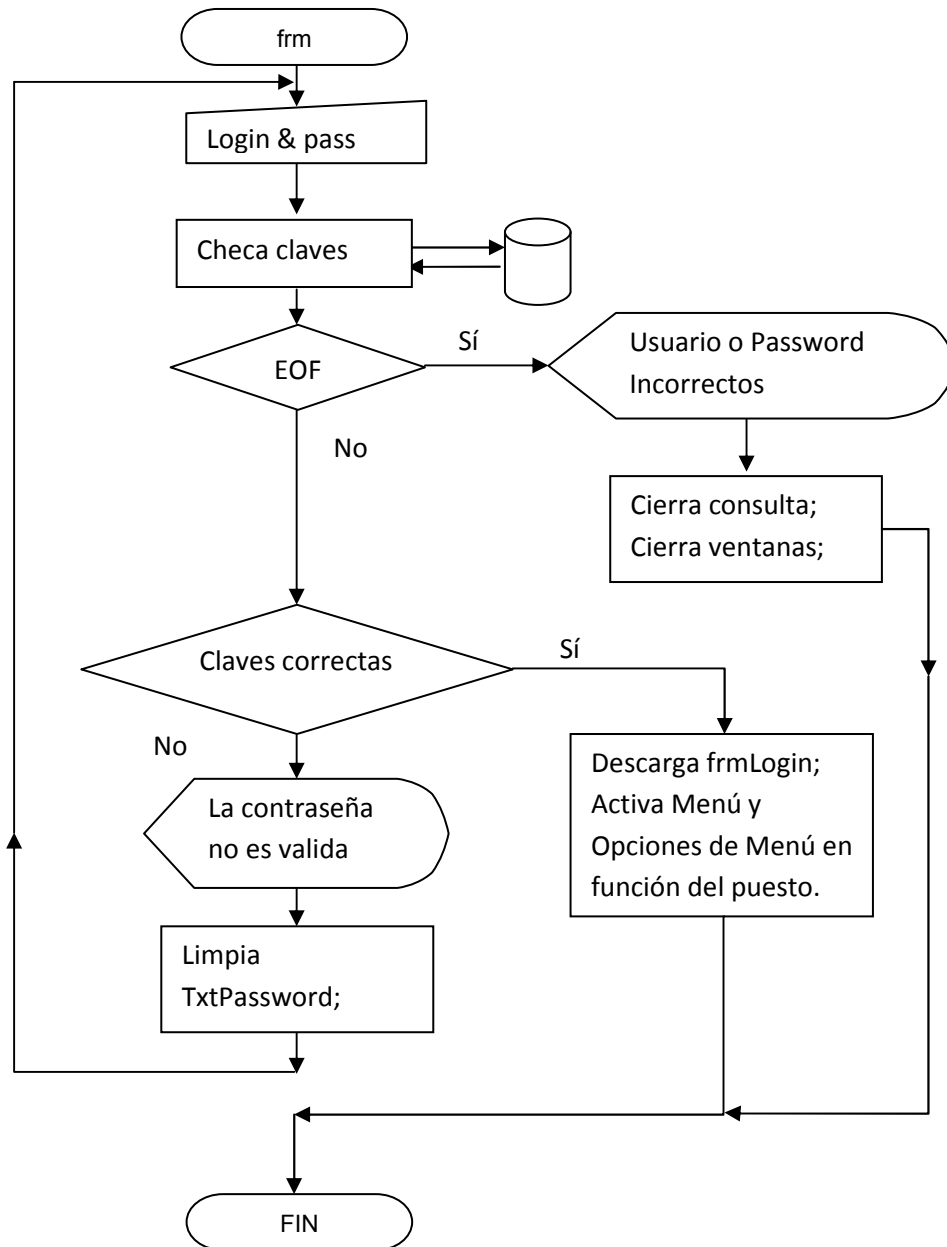
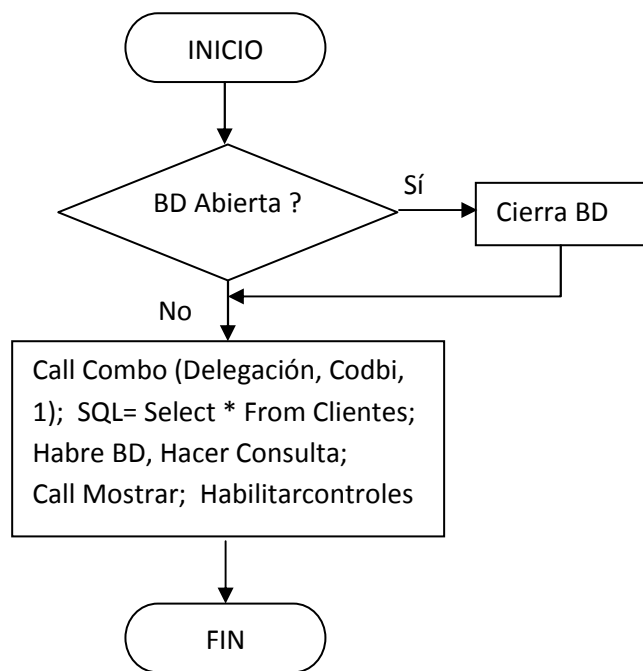


Figura 4.2.8.3 Diagrama de Flujo de la Subrutina frmLogin, para acceder al sistema.



El diagrama de flujo de la figura 4.2.8.3 del código de la forma frmLogin, muestra la búsqueda del login y password tecleados por el usuario. Si los encuentra y son correctas desactiva la forma activa; permite el acceso al sistema y, en función del login, activa o desactiva menús, submenús y ventanas del sistema. Si el password no es correcto envía mensaje y regresa a recaptura de datos. Si el login no existe se termina el proceso y se cierra el sistema.

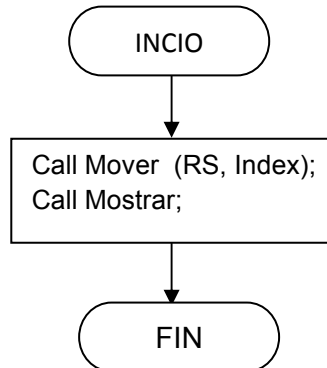
- **FRMCLI-FORM LOAD**



**Fig. 4.2.8.4. Diagrama de flujo de FRMLoad.**

Carga la forma FRMCLI. Verifica si la BD esta abierta; si lo está, la cierra y la reabre; si no la abre y se posiciona en el primer registro de clientes. Deshabilita controles, en función del cargo ocupado. La estructura es la misma, como la mostrada en la figura 4.2.8.4., para todas las formas de cada módulo del sistema.

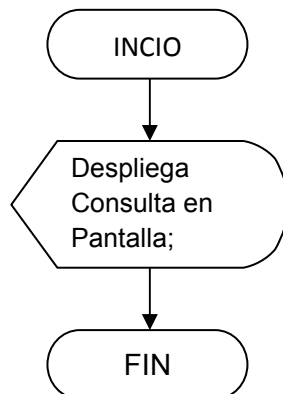
- **FRMCLI-CMDMOVER\_Click**



**Fig. 4.2.8.5. Diagrama de flujo de CMDMover\_Click.**

Procedimiento para moverse a través de la BD. Llama a Mover y luego a Mostrar. EL diagrama, como el mostrado en la figura 4.2.8.5., es igual para las formas FRMCTA-CMDMover y FRMEMP-CMDMover.

- **FRMCLI-MOSTRAR**



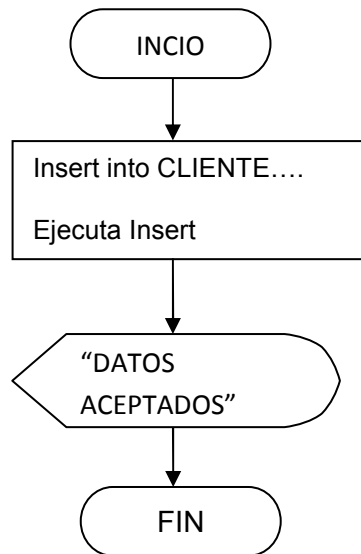
**Fig. 4.2.8.6. Diagrama de flujo de FRMCLI-Mostrar.**





Esta función permite al sistema desplegar los datos del cliente en pantalla. EL diagrama, como el mostrado en la figura 4.2.8.6., es igual para las formas FRMCTA-Mostrar y FRMEMP-Mostrar.

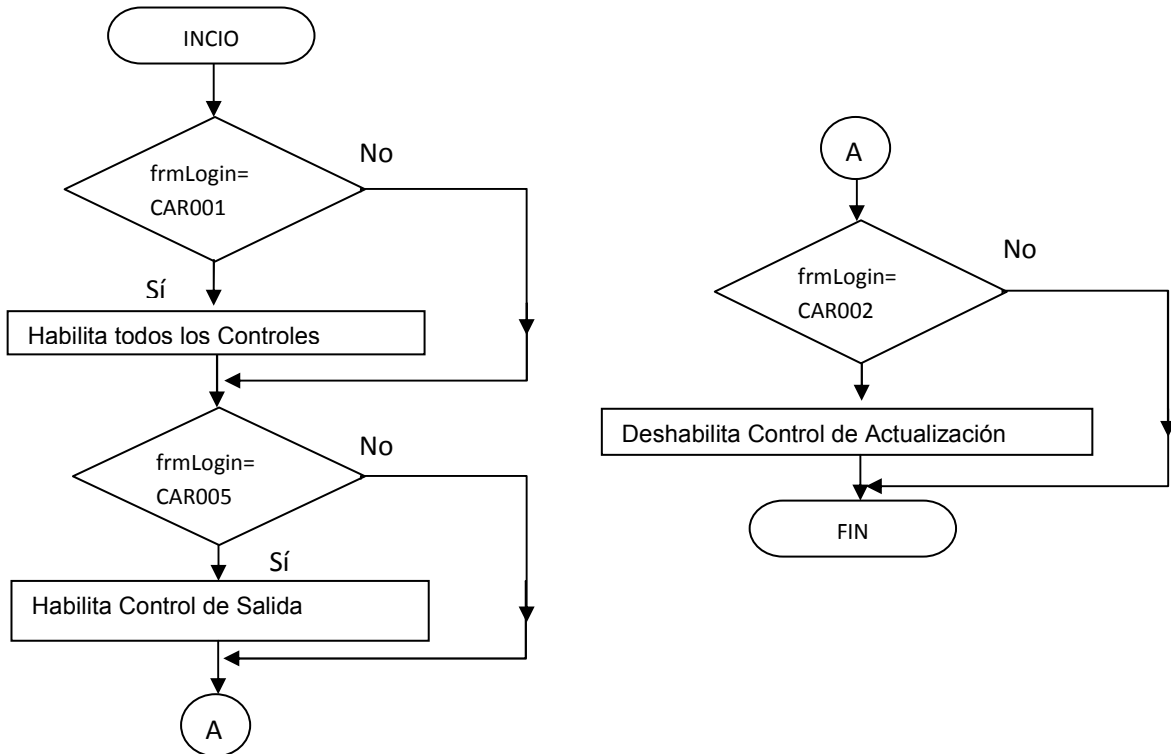
- **FRMCLI-Graba**



**Fig. 4.2.8.7. FRMCLI-Graba.**

La función Graba, permite al usuario incorporar nuevos clientes al sistema. Se arma la instrucción "Insert into cliente...", luego se ejecuta. Envía mensaje de aceptación. Esta función es similar en FRMCTA-Graba y FRMEMP-Graba y se muestra en la figura 4.2.8.7.

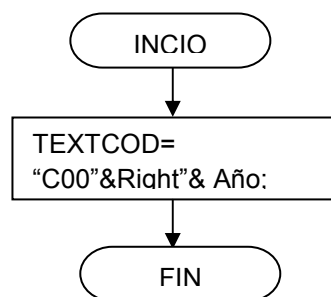
- **FRMCLI-Habilitacontroles**



**Fig. 4.2.8.8. Diagrama de flujo de HabilitaControles.**

Es la función que permite habilitar los controles a usar para manipular la BD; estos se activan en función del tipo de cargo. Esta función es similar en FRMCTA-Graba y FRMEMP-Graba Graba y se muestra en la figura 4.2.8.8.

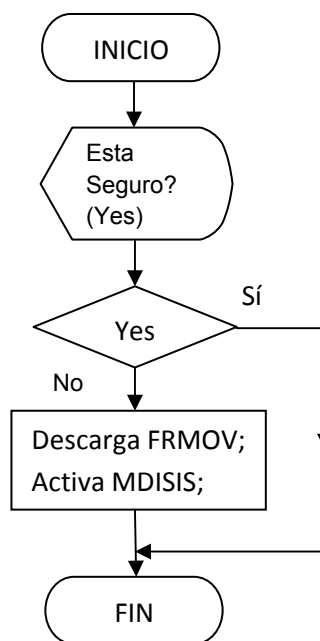
- **FRMCLI- GeneraCodigo**



**Fig. 4.2.8.9. Función Genera Código.**

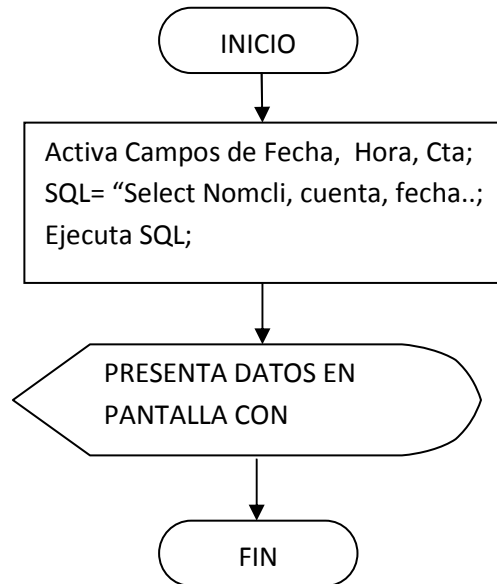
La función genera el código del nuevo usuario durante el alta de éste. Esta función es similar en las formas FRMCTA-Genera Código y FRMEMP- Genera Código, como se muestra en la figura 4.2.8.9.

- **FRMOV-CMDSALIR\_Click**



**Fig. 4.2.8.10. Diagrama de flujo de salida.**

Diagrama de flujo, para el código de salida, mostrado en la figura 4.2.8.10., del módulo de movimientos. Cuando se pulsa el botón, aparece aviso de interrogación y botones “Sí” y “No”. Si elige “Sí”, se descarga la forma de movimientos y se reactiva la pantalla principal. Si es “No”, continua en descarga FRMOV y activa MDISIS.

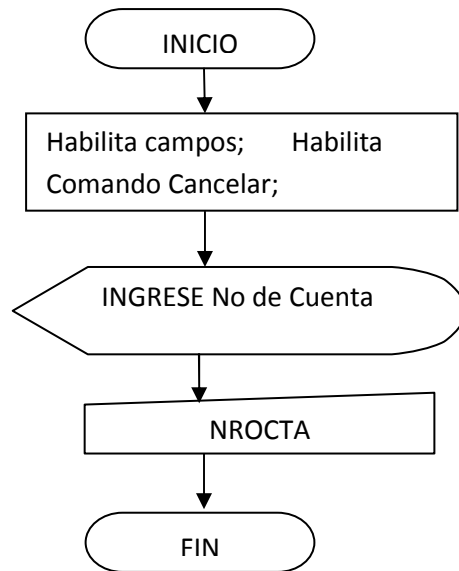
**FRMOV-CMDPRT\_Click**

**Fig. 4.2.8.11. Diagrama de flujo de FRMOV-CMDPRT\_Click.**

El diagrama de flujo, FRMOV-CMDPRT\_Click, que se muestra en la figura 4.2.8.11., describe el proceso de impresión de comprobante de movimiento registrado por el cajero.



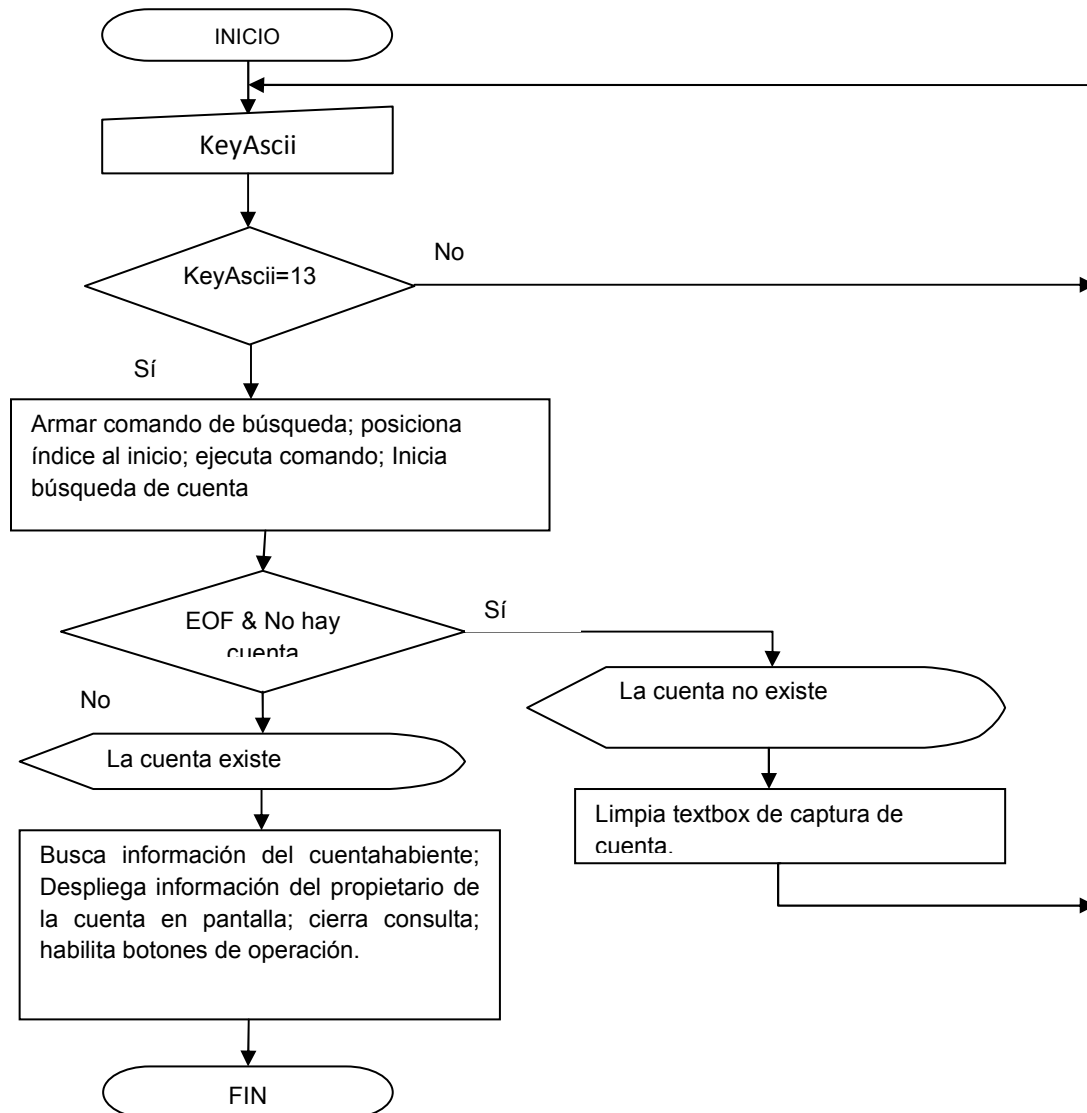
- **FRMOV-CMDIT\_Click**



**Fig. 4.2.8.12. Diagrama de flujo de subrutina CMDIT-Click.**

Diagrama de flujo, mostrado en la figura 4.2.8.12., para el código del botón de inicio del módulo de movimientos. Cuando se pulsa el botón, aparece la invitación a continuar y se activa el campo para captura de la cuenta del cliente.

- **FRMOV-TXTNC\_Keypress**

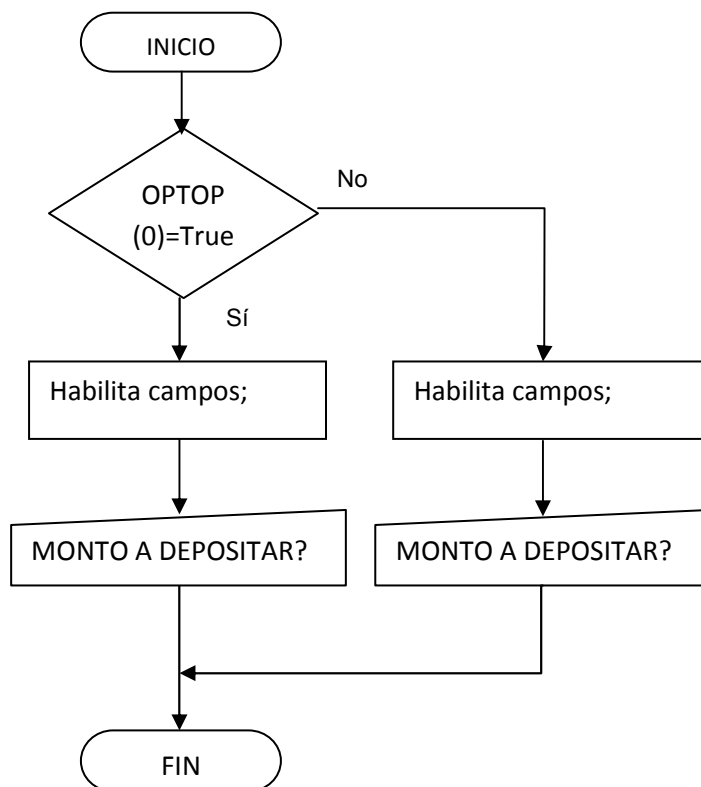


**Fig. 4.2.8.13. Diagrama de flujo de subrutina TXTNC.**

Una vez que se captura la cuenta del cliente, el sistema hace la búsqueda de esta. Si la encuentra despliega los datos del interesado y activa los botones de movimiento:

depósito y retiro, así como el de cancelación de operación. La figura 4.2.8.13, muestra el diagrama de flujo del código del textbox TXTNC.

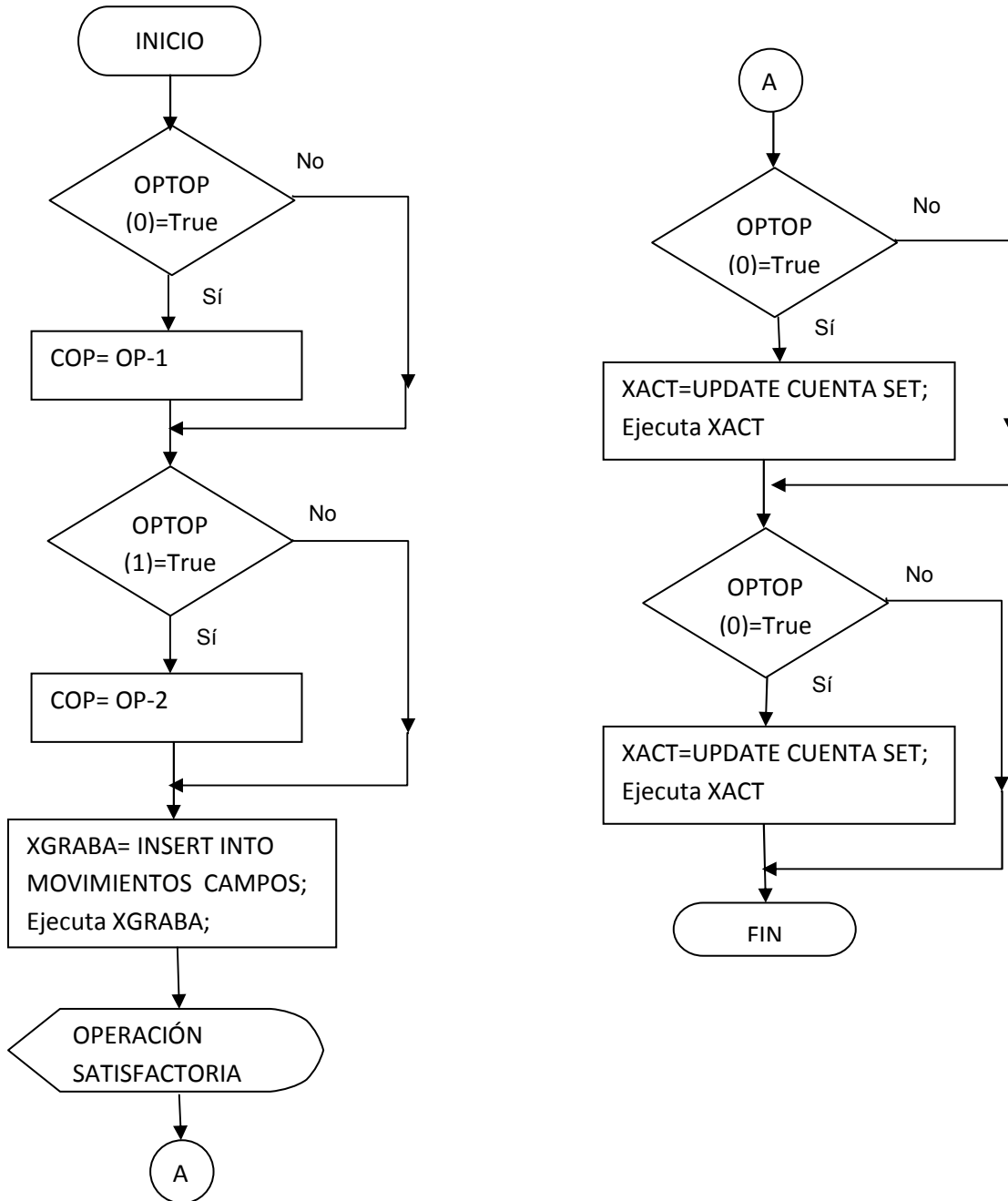
- **FRMOV- OPTOP\_Click**



**Fig. 4.2.8.14. Diagrama de flujo para rutina de depósito y retiro.**

El diagrama de flujo del código de los command buttons de “Retiro” y “Depósito”, mostrado en la figura 4.2.8.14., esquematiza el funcionamiento de estos; el cajero elige la opción deseada por el cuentahabiente; en ambos casos se habilita el textbox de captura del monto; debe capturarse el monto seguido por un “Enter” para continuar, como lo muestra el diagrama de flujo.

• **FRMOV-CMDREG\_Click**

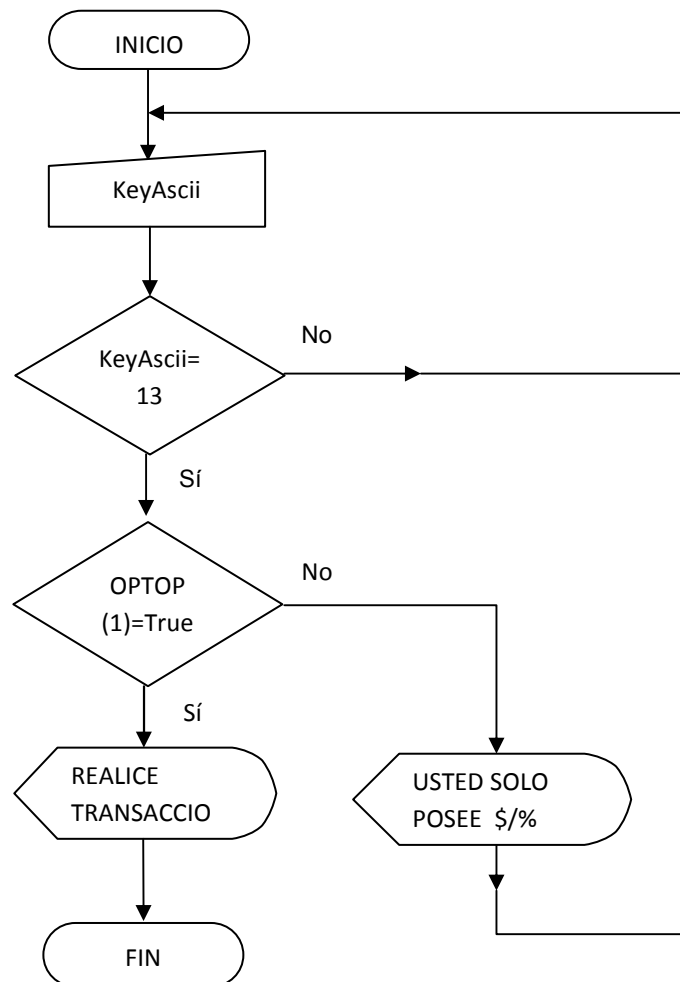


**Fig. 4.2.8.15. Diagrama de flujo para registro de operaciones.**



El diagrama de flujo, de la figura 4.2.8.15., esquematiza el funcionamiento del procedimiento de actualización de las operaciones de movimiento hechas por el operador en turno. Si la operación es de depósito, incrementa el saldo en la cuenta; si es retiro, decrementa el monto en la cuenta.

- **FRMOV-TXTMON\_KeyPress**

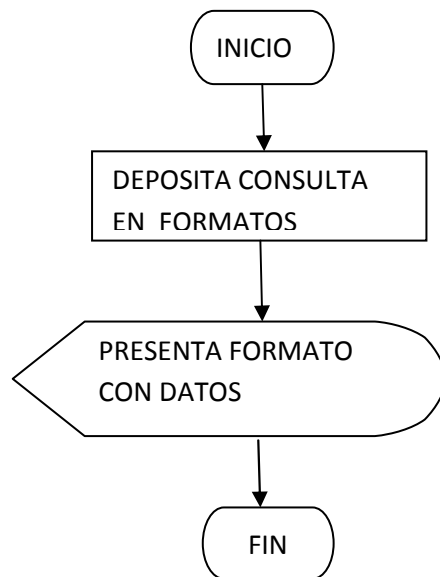


**Fig. 4.2.8.16. Diagrama de flujo para captura de monto en un movimiento.**



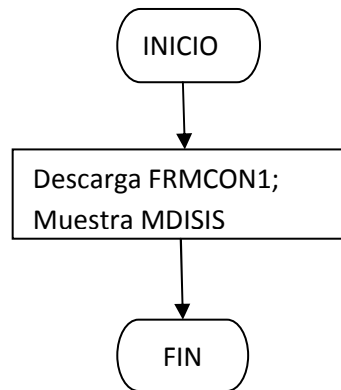
La figura 4.2.8.16, muestra el diagrama de flujo que esquematiza el funcionamiento del textbox TXTMON, que es donde se escribe el monto del movimiento de retiro, solicitado por el cliente, así como las acciones que siguen de acuerdo monto, es decir, si hace un retiro y este es mayor al saldo, el sistema envía un mensaje de advertencia y regresa a la ventana de captura. En caso contrario, avala la continuación de la operación.

- **FRMCON1-CMDIMPRIMIR\_Click**



**Fig. 4.2.8.17. Diagrama de flujo de Consultas.**

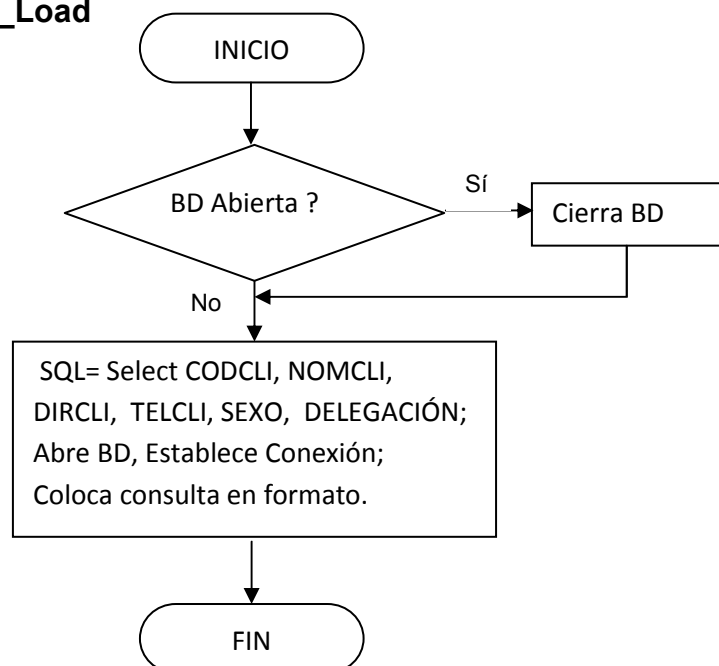
La forma FRMCON1 -CMDIMPRIMIR- , permite al usuario imprimir los datos estadísticos de movimientos consultados. El diagrama de flujo de la figura 4.2.8.17, muestra el proceso seguido para tal efecto. Los datos previamente obtenidos de la consulta –Form Load- se depositan en la forma especificada para ello, y después se muestran en pantalla. Después si lo usuario lo requiere, se imprimen. Esta misma estructura se aplica para el resto de las consultas FRMCON2, 3 y 4.

**FRMCON1-CMDSALIR**

**Fig. 4.2.8.18. Diagrama de flujo para quitar forma activa.**

La figura 4.2.8.18, muestra la simplicidad de la estructura del command button de salida de la forma FRMCON1, dedicada a la presentación de datos estadísticos de movimientos. Esta misma estructura es la misma para el resto de las formas del módulo de estadísticas.

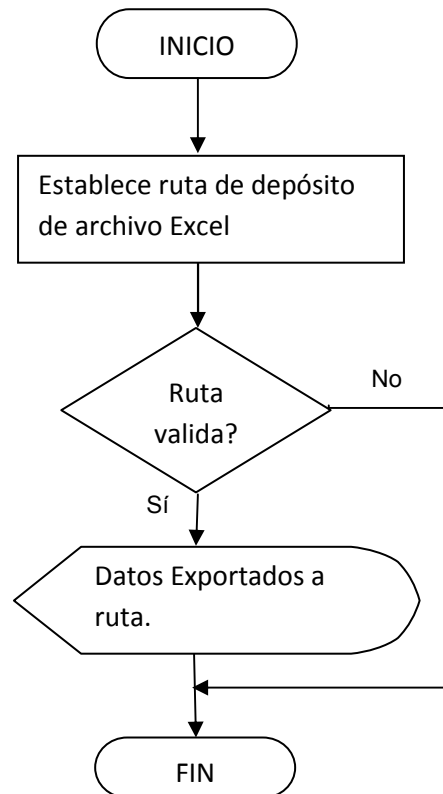
- **FRMCON1-Form\_Load**



**Fig. 4.2.8.19. Diagrama de flujo de Form\_Load.**

El proceso de consulta de información estadística de movimientos se inicia con la carga de la forma FRMCON1-Form Load. El diagrama de flujo, mostrado en la figura 4.2.8.19, indica el proceso de ejecución de instrucciones que permiten al usuario consultar en su totalidad la información de los movimientos realizados hasta ese momento. El proceso inicia verificando si la BD esta abierta; si es así, la cierra y después ejecuta la instrucción sql construida para tal efecto.

- **FRMCON1- CMD4XEXEX**



**Fig. 4.2.8.20. Diagrama de flujo para exportar datos a Excel.**

Para facilitar el manejo de la información y mantenerla disponible para efectos de análisis y consulta, se implementó un módulo de exportación a formato Excel. Como lo indica la figura 4.2.8.20., al oprimir el botón, el sistema verifica si la ruta establecida existe y de ser así deposita el archivo resultante de la exportación.

FRMDATEST1.

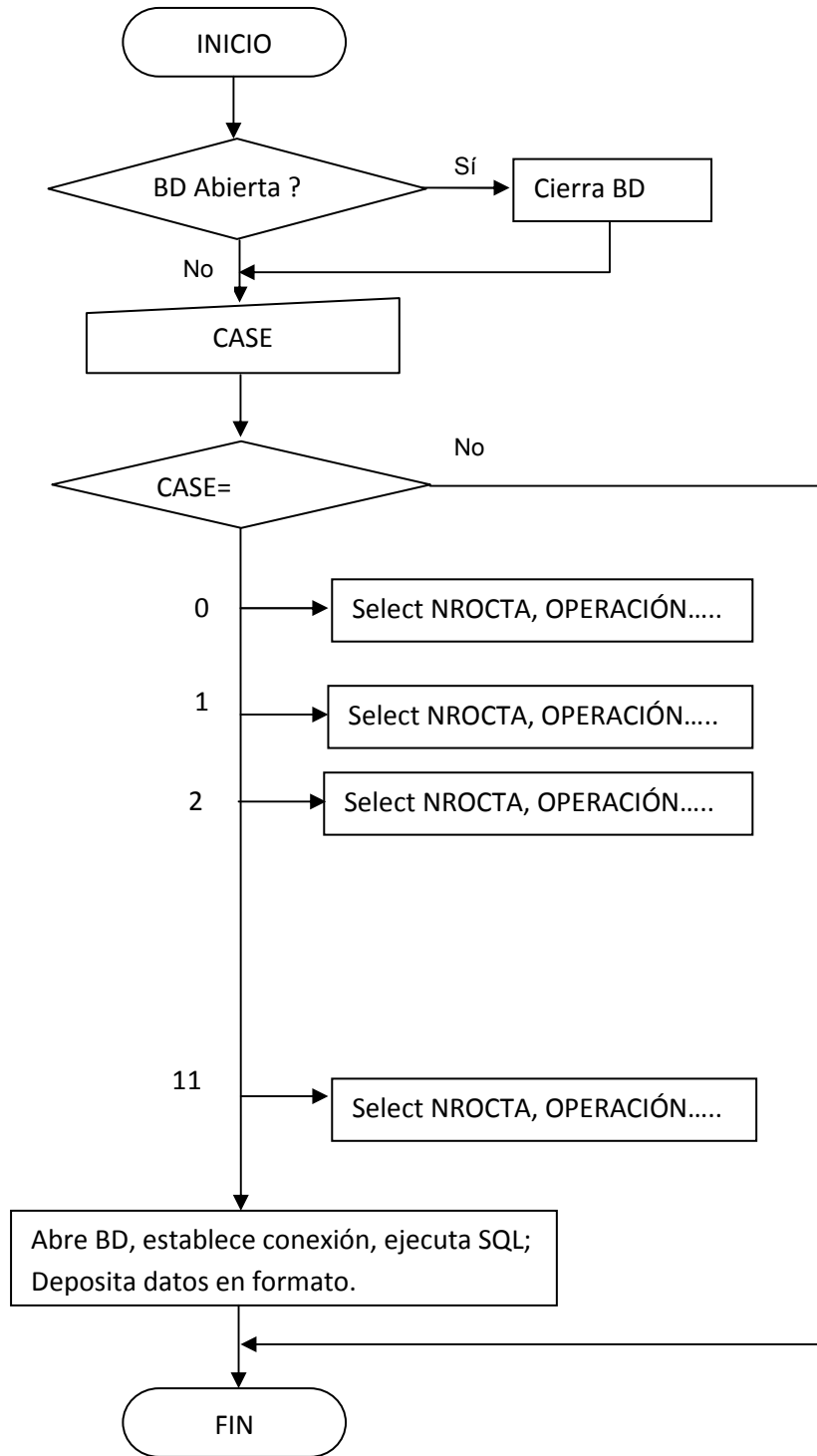


Fig. 4.2.8.21. Diagrama de flujo para consulta de operaciones por mes.

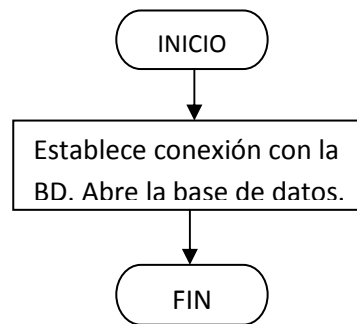


El diagrama de flujo de la figura 4.2.8.21., esquematiza el conjunto de instrucciones para obtener información de una operación en un mes seleccionado.

- **Module1 (Módulo Sistema Bancario)**

Las subrutinas presentadas en ésta sección permiten al sistema, en detalle, realizar funciones de búsqueda, apertura de la base de datos, moverse en la bases de datos, crear formatos, etc. Son funciones que son usadas por varios de los módulos del sistema.

- **BD**



**Fig. 4.2.8.22. Diagrama de flujo para conexión y apertura de la BD.**

La rutina representada en el diagrama de flujo de la figura 4.2.8.22 permite al sistema abrir la base de datos ubicada en una ruta previamente establecida.

• Mover

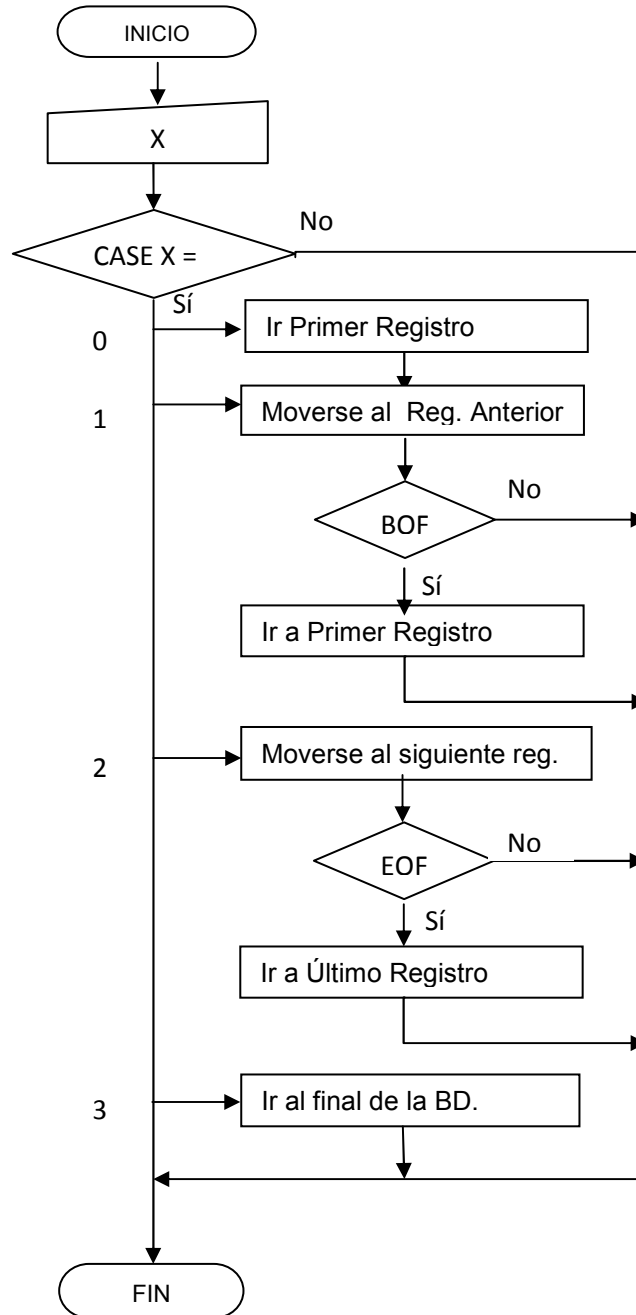
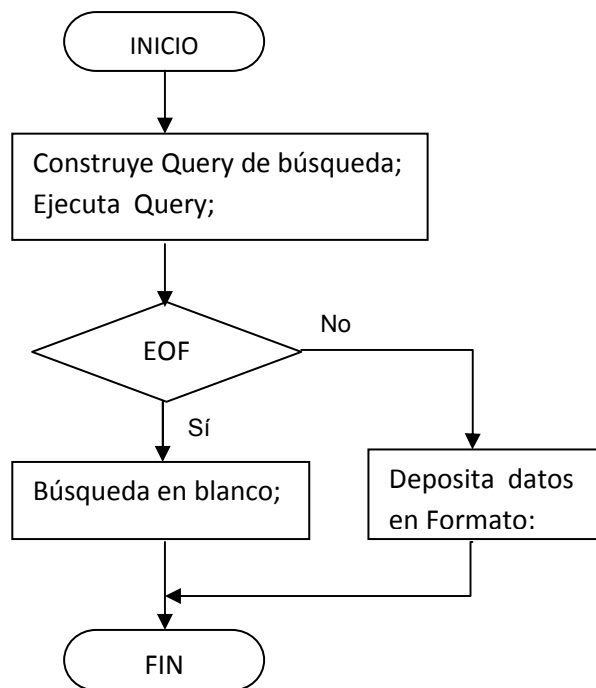


Fig. 4.2.8.23. Diagrama de flujo para usar botones de recorrido de BD.



La esquematización presentada en la figura 4.2.8.23., muestra el flujo de instrucciones que permiten al usuario utilizar los botones de movimiento en los módulos de usuarios, empleados y cuentas.

- **Buscar**

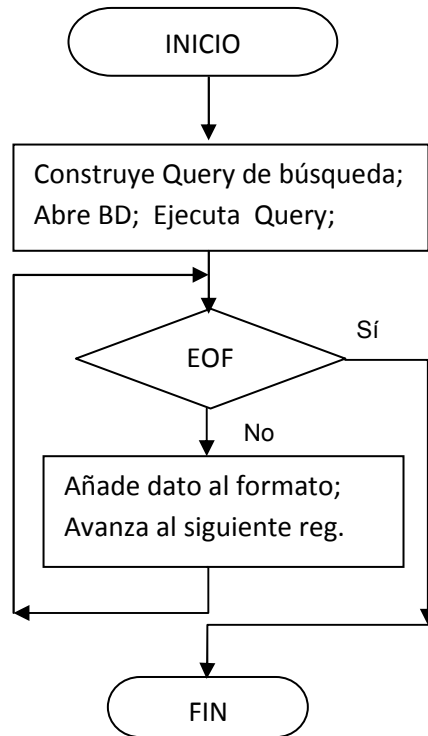


**Fig. 4.2.8.24. Diagrama de flujo de búsqueda de datos.**

La esquematización presentada en la figura 4.2.8.24., muestra la secuencia de instrucciones para la búsqueda de datos; Este módulo es utilizado para búsqueda datos de clientes, cuentas y empleados.



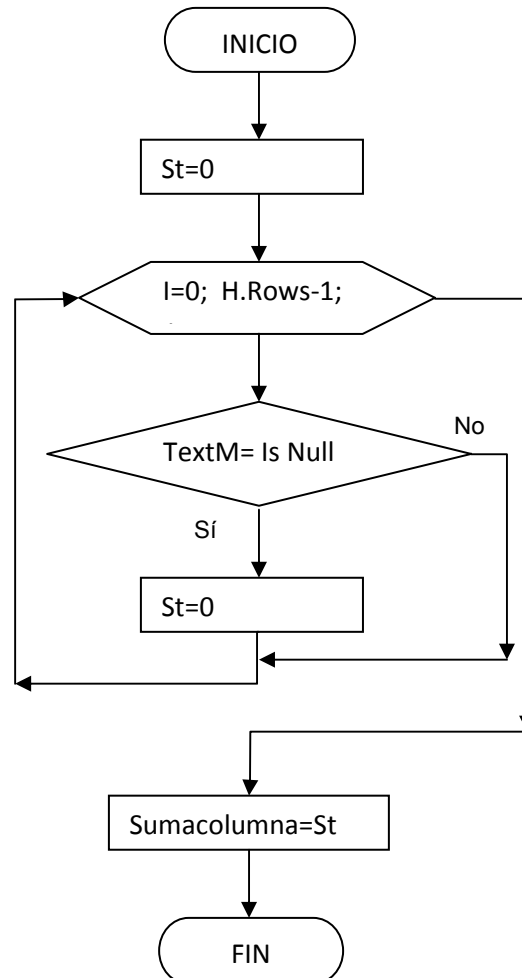
- **Combo**



**Fig. 4.2.8.25. Diagrama de flujo de búsqueda avanzada.**

El diagrama de flujo presentado en la figura 4.2.8.25, muestra la secuencia de instrucciones para la búsqueda de datos en el submenú “búsqueda avanzada”.

- **Sumacolumna**



**Fig. 4.2.8.26. Diagrama de flujo para suma de columna de datos.**

El diagrama de flujo, en la figura 4.2.8.26., muestra la secuencia de instrucciones para Sumar los datos en las opciones de “Datos Estadísticos”, Como resumen de movimientos por mes, por tipo de cuenta, etc.

### Exportar\_Excel

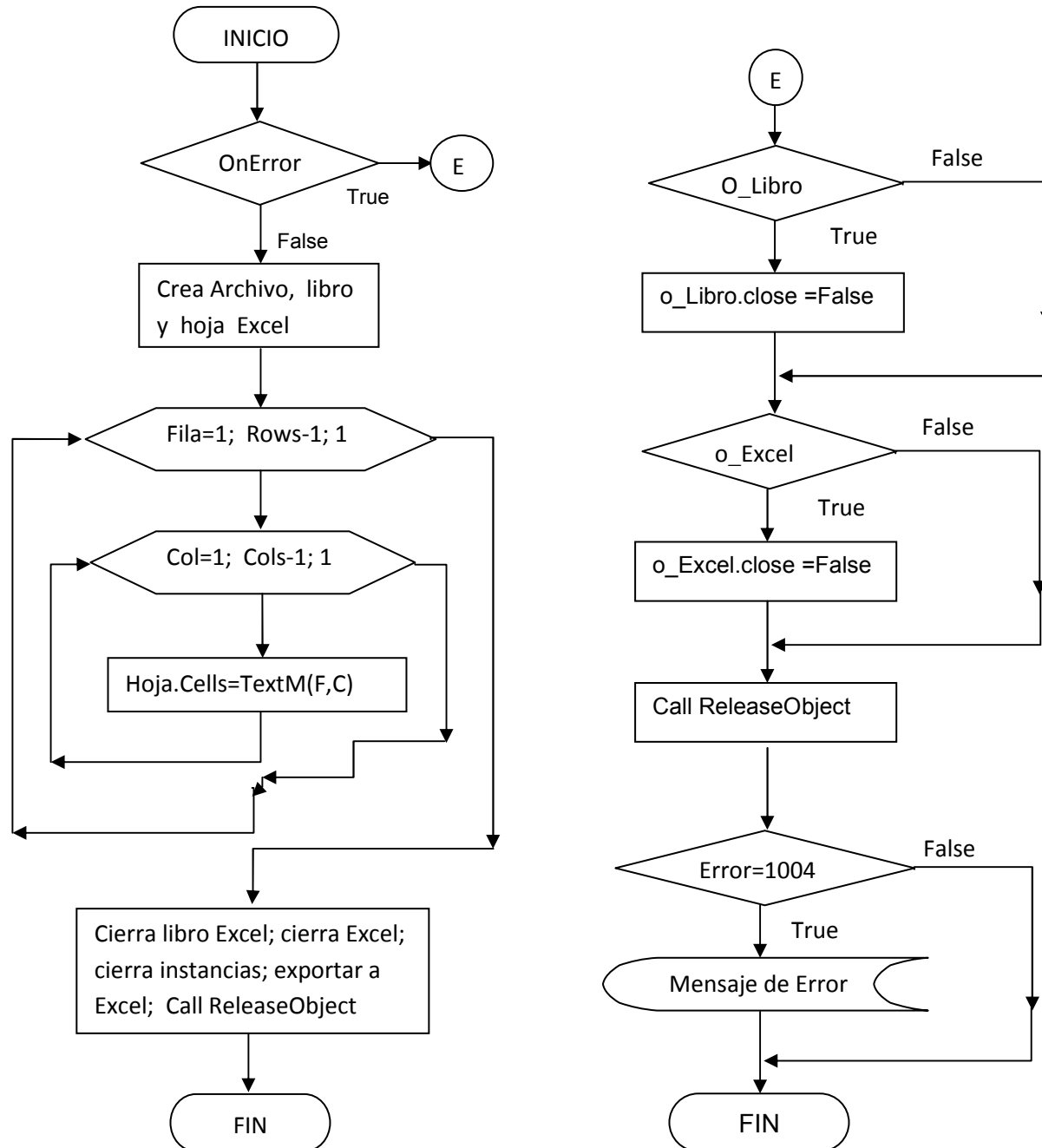


Fig. 4.2.8.27. Diagrama de flujo para proceso de exportar datos a Excel.



La figura 4.2.8.27, muestra la secuencia de instrucciones exportar reportes de datos estadísticos a Excel.

#### **4.2.9 Diccionario de Datos.**

Un diccionario de datos es un conjunto de metadatos que contiene las características lógicas de los datos que se van a utilizar en el sistema que se programa, incluyendo nombre, descripción, alias, contenido y organización.

Estos diccionarios se desarrollan durante el análisis de flujo de datos y ayuda a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño del proyecto.

Identifica los procesos donde se emplean los datos y los sitios donde se necesita el acceso inmediato a la información, se desarrolla durante el análisis de flujo de datos y auxilia a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño.

Este documento es de vital importancia para tener registro de cómo están diseñadas las bases de datos, incluye información de diseño como, nombre de la tabla, y detalles de la composición como: nombre de campo, tipo de dato, tamaño de campo, además de una descripción de cada campo acerca del dato que va a representar y las relaciones que tienen cada una de ellas por los campos en común con las diferentes tablas.



Debe estar diseñado de una forma que pueda ser comprensible para cualquier persona que necesita consultar dicho diccionario véase (tablas 4.2.9.1, 4.2.9.2, 4.2.9.3, 4.2.9.4, 4.2.9.5, 4.2.9.6, 4.2.9.7, 4.2.9.8, 4.2.9.9).

**Nombre de la Tabla:** Cargos

**Fecha de Creación:** 14/04/2011

**Descripción:** Tabla que contendrá los cargos de los usuario, así como las claves de acceso.

Acrónimo	Tipo	Tamaño	Título	Permite longitud cero	Es campo llave	Descripción
CODCAR	Texto	50	código	si	si	Código de cargo
DESCAR	Texto	50	cargo	si	no	Descripción del cargo

**Relaciones:**

Se relaciona con la tabla de empleados

**Campos clave:**

CODCAR

**Tabla de cargos 4.2.9.1**

**Nombre de la Tabla:** Clientes**Fecha de Creación:** 14/04/2011**Descripción:** Tabla que contendrá los códigos de los clientes así como su información personal.

<b>Acrónimo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Título</b>	<b>Permite longitud cero</b>	<b>Es campo llave</b>	<b>Descripción</b>
CODCLI	Texto	50	código	si	si	Código del cliente
NOMCLI	Texto	50	nombre	si	no	Nombre del cliente
DIRCLI	Texto	50	dirección	si	no	Dirección del cliente
TELCLI	Texto	50	teléfono	si	no	Teléfono del cliente
SEXO	Texto	50	Sexo	si	no	Genero
CODDIS	Texto	50	distrito	si	Si	Es el código del distrito al que pertenecen

**Relaciones:**

Se relaciona con las tablas de cuentas y de delegaciones

**Campos clave:**

CODCLI Y CODIS

**Tabla de clientes 4.2.9.2**



**Nombre de la Tabla:** Cuenta **Fecha de Creación:** 14/04/2011

**Descripción:** Tabla que contendrá las diferentes cuentas de ahorro, códigos de clientes y empleados.

Acrónimo	Tipo	Tamaño	Título	Permite longitud cero	Es campo llave	Descripción
NROCTA	Texto	50	Número de cuenta	Si	no	Es el número de cuenta que tienen los clientes
CODCLI	Texto	50	Cliente	Si	si	Código del cliente
CODEMP	Texto	50	Empleado	Si	si	Código del empleado
CODMON	Texto	50	Moneda	Si	si	Código de la moneda que se va a utilizar
CODTCTA	Texto	50	Cuenta	Si	si	Genero
MONTO	Número	Entero largo	Monto	No	no	Es el monto que se desea retirar o depositar.
FECHAREG	Fecha/Hora	Fecha corta	Fecha de registro	No	no	Es la fecha en la que se realiza el registro.

**Relaciones:** Se relaciona con las tablas de clientes, empleados, Y Movimientos, moneda y tipo de cuenta.

**Campos clave:** CODCLI, CODEMP, CODMON, CODTCTA.

**Tabla de Cuenta 4.2.9.3**

**Nombre de la Tabla:** Delegaciones **Fecha de Creación:** 14/04/2011

**Descripción:** Tabla que contendrá todos los delegaciones.

Acrónimo	Tipo	Tamaño	Título	Permite longitud cero	Es campo llave	Descripción
CODIS	Texto	50	código	si	Si	Código de distrito
DISTRITO	Texto	50	distrito	si	No	Nombre del distrito

**Relaciones:** Se relaciona con las tablas de clientes y empleados

**Campos clave:** CODIS

**Tabla de Delegaciones 4.2.9.4**

**Nombre de la Tabla:** Empleados**Fecha de Creación:** 14/04/2011**Descripción:** Tabla que contendrá los diferentes empleados que laboran, así como su información personal.

Acrónimo	Tipo	Tamaño	Título	Permite longitud cero	Es campo llave	Descripción
CODEMP	Texto	50	código	si	si	Codigo Empleado
EMPLEADO	Texto	50	nombre	si	no	Nombre del Empleado
CODCAR	Texto	50	cargo	si	si	Código del empleado
DIR	Texto	50	dirección	si	no	Dirección del empleado
TEL	Texto	50	teléfono	si	no	Teléfono del Empleado
SEXO	Texto	50	sexo	Si	no	Genero
CODIS	Texto	50	distrito	si	no	Es el código del Distrito
PASSWORD	Texto	10	clave	si	no	Clave de usuario del empleado

**Relaciones:**

Se relaciona con las tablas de delegaciones, cargos y cuenta.

**Campos clave:**

CODEMP, CODCAR Y CODIS.

**Tabla de Empleados 4.2.9.5****Nombre de la Tabla:** Monedas**Fecha de Creación:** 14/04/2011**Descripción:** Tabla que contendrá los tipos de moneda que se manejaran.

Acrónimo	Tipo	Tamaño	Título	Permite longitud cero	Es campo llave	Descripción
CODMON	Texto	50	Código de moneda	si	si	Código de tipo de moneda
DESMON	Texto	50	descripción	si	no	Descripción del tipo de moneda a utilizar

**Relaciones:**

Se relaciona con la tabla de cuenta

**Campos clave:**

CODMON

**Tabla de Monedas 4.2.9.6**





**Nombre de la Tabla:** Movimientos **Fecha de Creación:** 14/04/2011

**Descripción:** Tabla que contendrá los movimientos, los montos y las fechas de registro.

Acrónimo	Tipo	Tamaño	Título	Permite longitud cero	Es campo llave	Descripción
NROCTA	Texto	50	Número de cuenta	si	si	Número que se le da a las cuentas
CODOPE	Texto	50	Código de operación	si	si	Nos indica el tipo de operación
MONTO MOV	Numero	Entero largo	monto		no	Monto de dinero depositado o retirado
FECHA MOV	Fecha/hora	Fecha corta	Fecha de movimiento		no	Registro de la fecha en que se hizo la operación
HORA MOV	Fecha/hora		Hora de movimiento		no	Registro de la hora en que se realizo la operación

**Relaciones:** Se relaciona con las tablas de cuenta y opbancaria **Campos clave:** CODOPE

**Tabla de Movimientos 4.2.9.7**

**Nombre de la Tabla:** Opbancaria **Fecha de Creación:** 14/04/2011

**Descripción:** Tabla que contendrá los tipos de movimientos.

Acrónimo	Tipo	Tamaño	Título	Permite longitud cero	Es campo llave	Descripción
CODOPE	Texto	50	Código de operación	si	si	Código de distrito
DESOPE	Texto	50	descripción	si	no	Nombre del distrito

**Relaciones:** Se relaciona con la tabla de movimientos **Campos clave:** CODOPE

**Tabla de Opbancaria 4.2.9.8**



**Nombre de la Tabla:** Tipo de cuenta **Fecha de Creación:** 14/04/2011

**Descripción:** Tabla que contendrá los diferentes tipos de cuentas.

<b>Acronimo</b>	<b>Tipo</b>	<b>Tamaño</b>	<b>Titulo</b>	<b>Permite longitud cero</b>	<b>Es campo llave</b>	<b>Descripción</b>
CODTCTA	Texto	50	Código t.c.	si	si	Código que se le asigna a los diferentes tipos de cuenta
DESTCTA	Texto	50	descripción	si	no	Se describe los diferentes tipos de cuenta

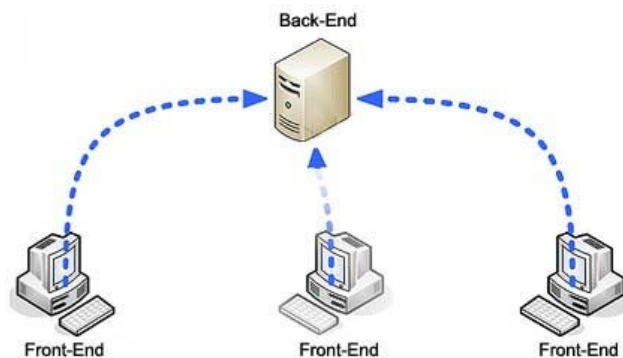
**Relaciones:**  
Se relaciona con la tabla de cuenta

**Campos clave:**  
CODTCTA

**Tabla de Tipo de Cuenta 4.2.9.9**

### 4.3 DISEÑO DE LA BASE DE DATOS

El back-end es la parte de la aplicación que nos permitirá determinar cómo se almacenará nuestra información, la forma en la cual tendremos organizadas las tablas y tiene por objetivo crear la estructura del sistema modular para facilitar la captura, búsqueda, actualización y manipulación de los datos. El back-end se encuentra en un computadora que funciona como servidor, dentro de una carpeta compartida y la cual solamente contiene las tablas, a su vez el front-end, del cual hablaremos más adelante, puede estar en local en cada ordenador y los usuarios tendrán accesos directos instalados en su estación u ordenador. Esta base tendrá todos los objetos para la exploración, edición y creación de archivos, es decir formularios, consultas, macros, informes, pero no contendrá ninguna tabla en local; a esta base se vincularán las tablas del back-end. Véase figura 4.3.1.



**Figura 4.3.1 El Back-end y Front-end**

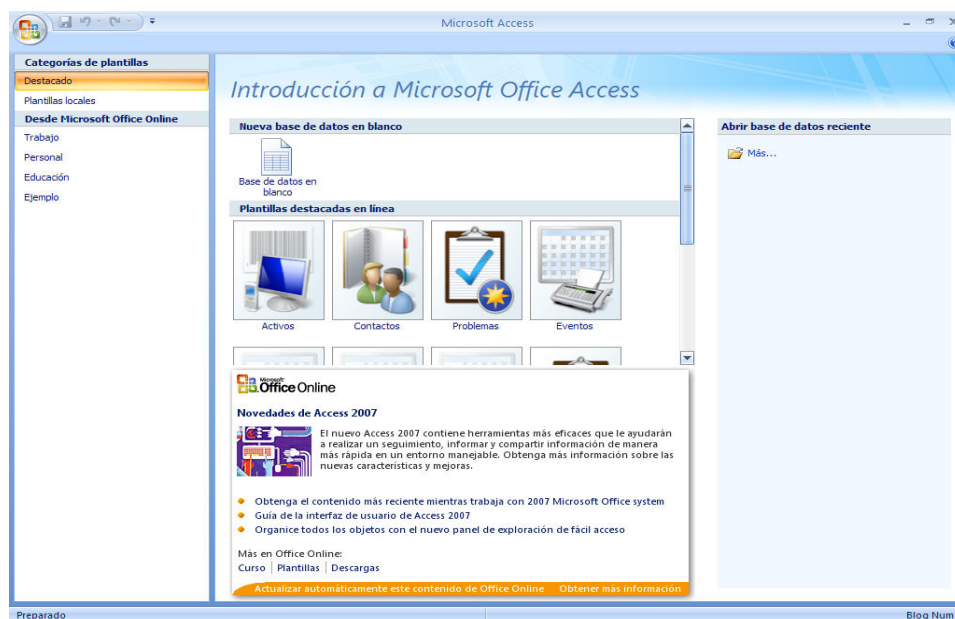
Una base de datos suele definirse como un conjunto de datos relacionados y organizados sistemáticamente. Tiene como fin almacenar la información evitando la redundancia de datos y permitiendo que estos se puedan recuperar de forma rápida, segura, que sean verídicos y que al ser modificados no se afecte la integridad de la base.

Los DBMS son un conjunto de programas que corren en el sistema operativo como un gran y sofisticado programa de aplicación. La construcción de la base de datos del sistema depende de las características propias del DBMS y del modelo de datos seleccionado para el sistema.

En el proceso del control de la información, perteneciente a la etapa del diseño, se definirán las tablas, así como la estructura de los datos, utilizando para este fin Microsoft Access.

El sistema de bases de datos para este proyecto está formado por varias tablas que mantienen las relaciones que se especificaron en el diccionario de datos. Los datos contenidos en cada una de las tablas son manipulados por el front-end, desarrollado en Visual Basic 6. En la base de datos no se realizan validaciones, debido a que éstas se llevan a cabo en el front-end.

Para crear la base de datos abrimos Microsoft Access, en Inicio → Programas → Microsoft Office → Microsoft Access. Se despliega la pantalla inicial. Véase figura 4.3.2.



**Figura 4.3.2 Pantalla inicial Access 2007.**

Se decidió crear nueva base de datos ya que ninguna de las prediseñadas es adecuada a las necesidades del cliente. Por lo anterior, elegimos la opción Nueva Base de Datos en blanco. Introducimos el nombre de nuestra base de datos, que en nuestro caso nombramos como Sistema Bancario, elegimos la ruta en donde deseamos guardarla y damos click en crear. Véase figura 4.3.3.

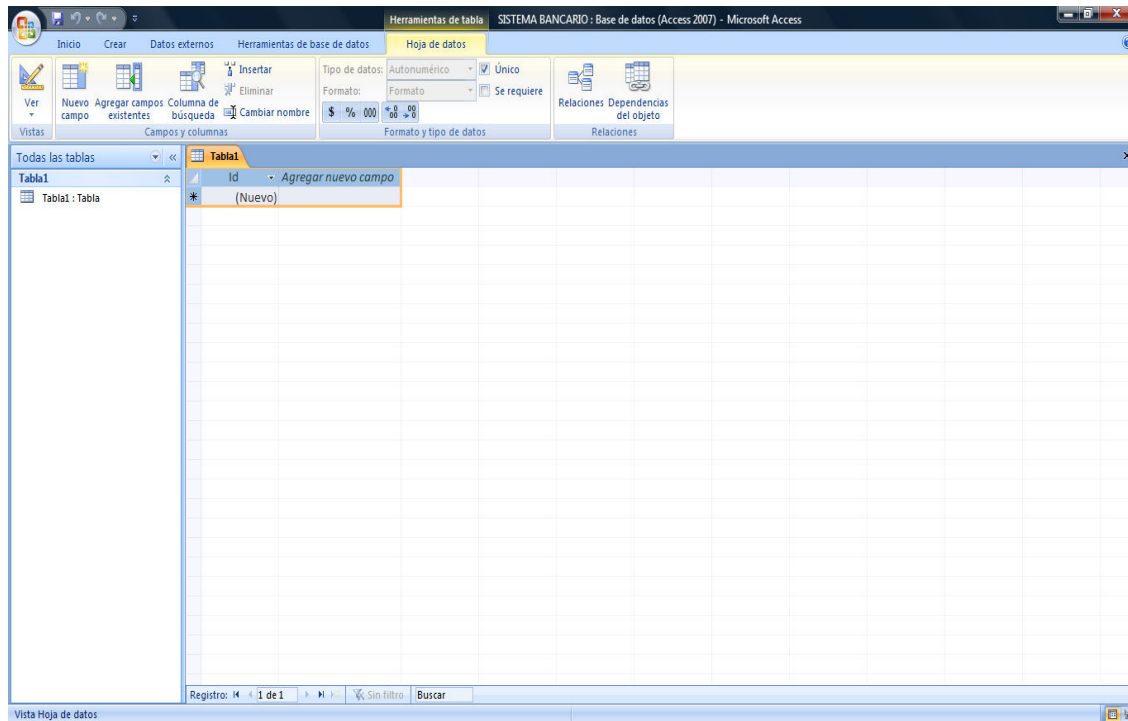


**Figura 4.3.3. Creación de Nueva Base Datos.**

Automáticamente Access abrirá una nueva tabla en la cual comenzaremos a llenar los campos. A la izquierda aparece el Panel de Exploración, desde donde más adelante, podremos seleccionar todos los objetos que sean creados dentro de la base de datos. Véase figura 4.3.4.

Algo que se tomó en cuenta antes de crear la base de datos en Access, fue el bosquejo del diseño, modelo entidad relación, para que al momento de pasar el diccionario de datos fuera fácil crear cada una de las tablas que conforman a la base de datos, y

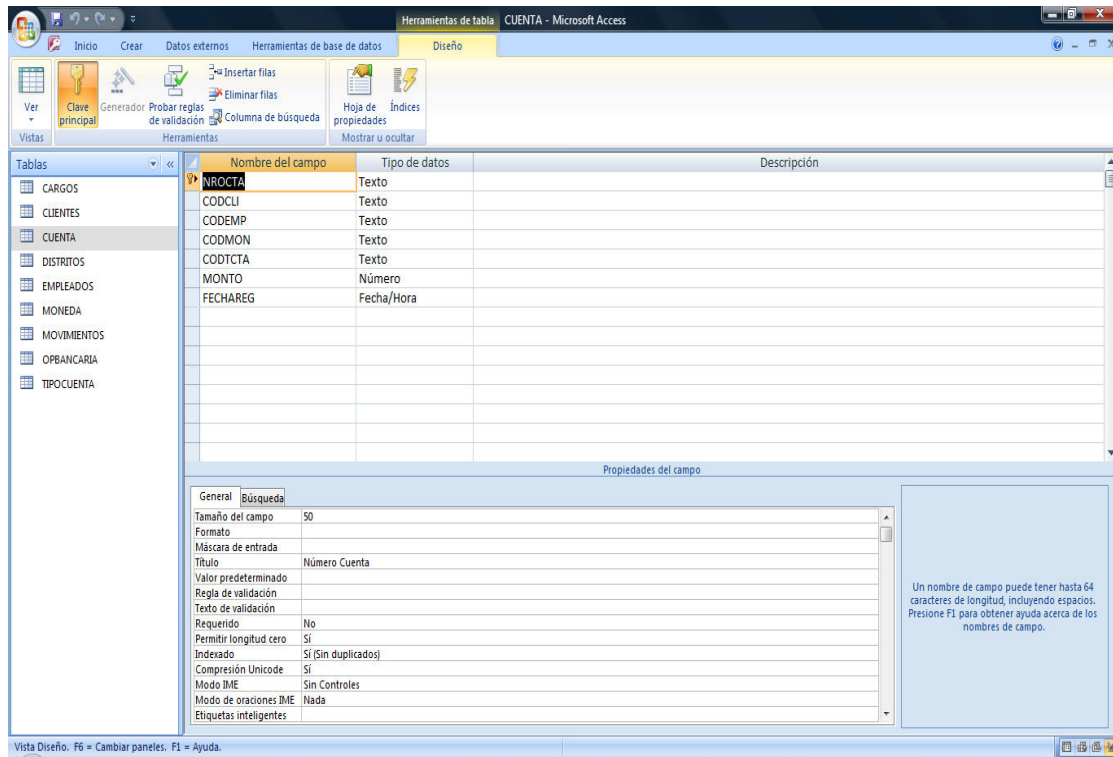
definir cuáles serían los campos que serían los índices para que el sistema sea eficiente y adecuado.



**Figura 4.3.4. Creación de las Tablas.**

Para crear las tablas del sistema utilizamos la opción “Vista de Diseño”. Haciendo click en la pestaña “Crear”, para visualizar sus opciones y en el grupo “Tablas” seleccionamos “Diseño de Tabla”, aparecerá la Vista de diseño de la tabla.

En la pestaña aparece el nombre de cada tabla. A continuación ingresamos los datos en columnas (campos) y filas (registros) que cada campo contiene: Nombre del campo, Tipo de datos y Descripción. En la parte inferior tenemos, a la izquierda, dos pestañas (General y Búsqueda) para definir propiedades del campo, es decir características adicionales de la columna que estamos definiendo. Las propiedades se agrupan en dos pestañas, la pestaña “General” donde indicamos las características generales del campo y la pestaña “Búsqueda” en la que podemos definir una lista de valores válidos para el campo. Véase fig. 4.3.5.



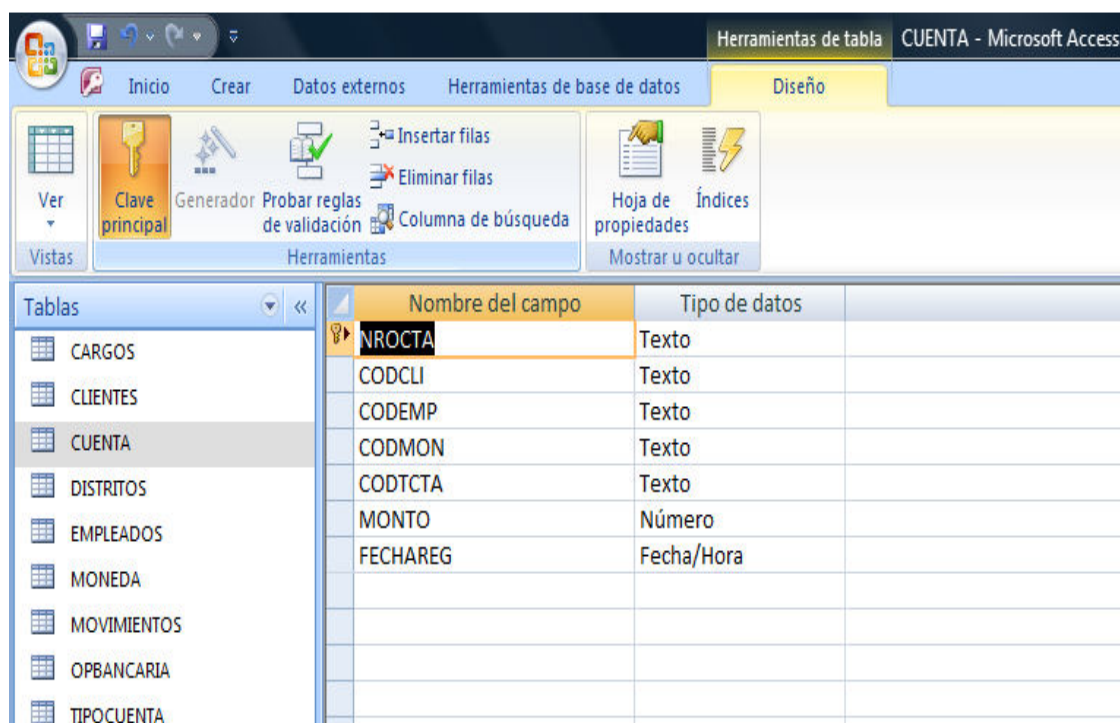
**Figura 4.3.5. Ingreso de datos en las Tablas.**

Antes de guardar cada una de las tablas, asignamos una clave principal. La clave principal proporciona un valor único para cada fila de la tabla y nos sirve de identificador de registros, de tal forma que con esta clave podemos identificar, sin ningún tipo de equivocación, el registro al cual nos referimos. No podemos definir más de una clave principal, pero podemos tener una clave principal compuesta por más de un campo.

Para definir un campo como clave principal, se ejecutan los siguientes pasos:

- Hacer click sobre el campo que será clave principal.
- Hacer click sobre el botón Clave principal en el marco Herramientas de la pestaña Diseño.

A la izquierda del nombre del campo aparecerá una llave indicándonos que dicho campo es la clave principal de la tabla. Véase figura 4.3.6.



**Figura 4.3.6. Asignación de la Clave Principal.**

Para la elaboración de nuestra base de datos requerimos llenar las tablas correspondientes a Clientes, Cuentas, Empleados, Cargos, Delegaciones, Tipo de Moneda, Operación Bancaria, Tipo de cuenta y Movimientos, siguiendo el procedimiento descrito anteriormente para el llenado de los datos en cada una de las tablas de nuestro sistema y la asignación de la clave principal en cada caso.

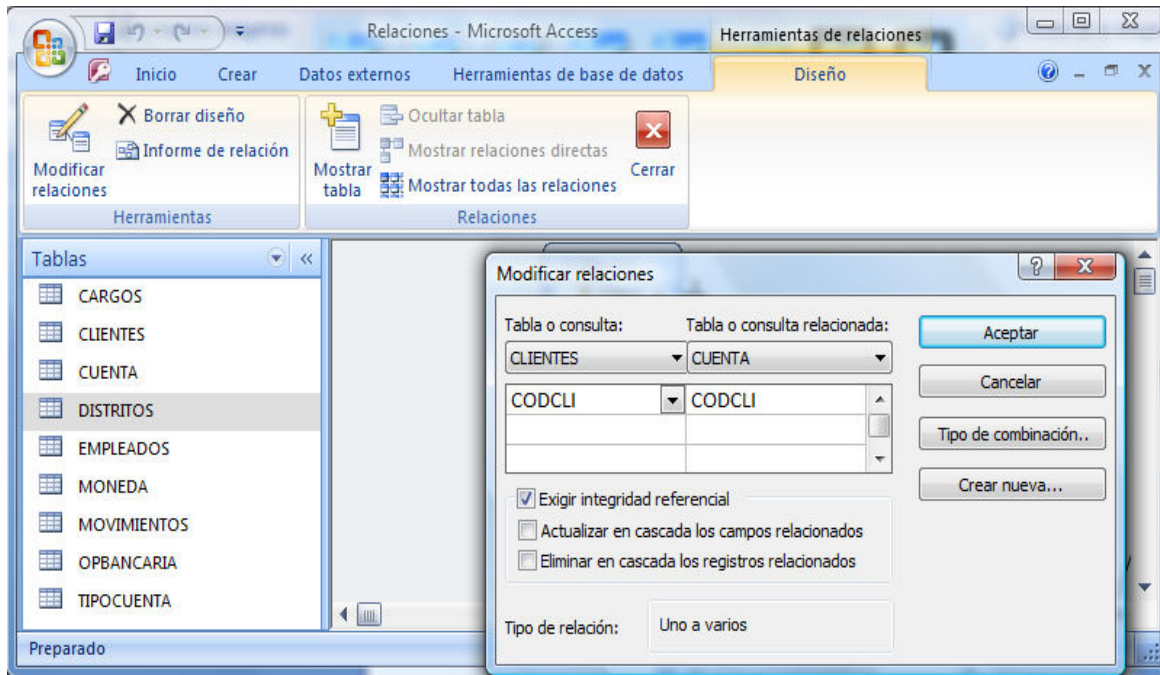
Para crear relaciones en Access, primero deberemos acceder a la ventana Relaciones, hacemos click en el botón Relaciones, que se encuentra en la pestaña Herramientas de base de datos.





En esta parte seleccionamos cada una de las tablas que pertenecen a la relación haciendo clic sobre ella; aparecerá dicha tabla –sombreada- y después hacemos click sobre el botón Agregar.

Repetimos los dos pasos anteriores hasta añadir todas las tablas de las relaciones que necesitamos crear para nuestro sistema. Para crear la relación vamos sobre el campo de relación de la tabla principal, pulsamos el botón izquierdo del ratón y manteniéndolo pulsado arrastramos hasta el campo de la tabla secundaria. Siempre deben ser campos que contengan el mismo tipo de información y por lo tanto del mismo tipo. Observamos en la parte inferior el tipo de relación que se asignará dependiendo de las características de los campos de relación (en nuestro caso uno a varios). Activamos el recuadro “Exigir integridad referencial” haciendo click sobre éste. La integridad referencial es un sistema de reglas que utiliza Access para asegurarse que las relaciones entre registros de tablas relacionadas son válidas y que no se borren o cambien datos relacionados de forma accidental. Al exigir integridad referencial en una relación le estamos diciendo a Access que no nos deje introducir datos en la tabla secundaria si previamente no se ha introducido el registro relacionado en la tabla principal. En nuestro caso activamos las casillas Actualizar en cascada los campos relacionados y Eliminar en cascada los registros relacionados. De esta forma, indicamos que cuando se cambie el valor del campo de la tabla principal, automáticamente cambiarán los valores de sus registros relacionados en la tabla secundaria y cuando se elimina un registro de la tabla principal se borrarán también los registros relacionados en la tabla secundaria. Véase figura 4.3.7.



**Figura 4.3.7. Creación de Relaciones.**

Para finalizar la relación, hacemos click sobre el botón Crear. Se creará la relación y ésta aparecerá en la ventana Relaciones. Véase figura 4.3.8.

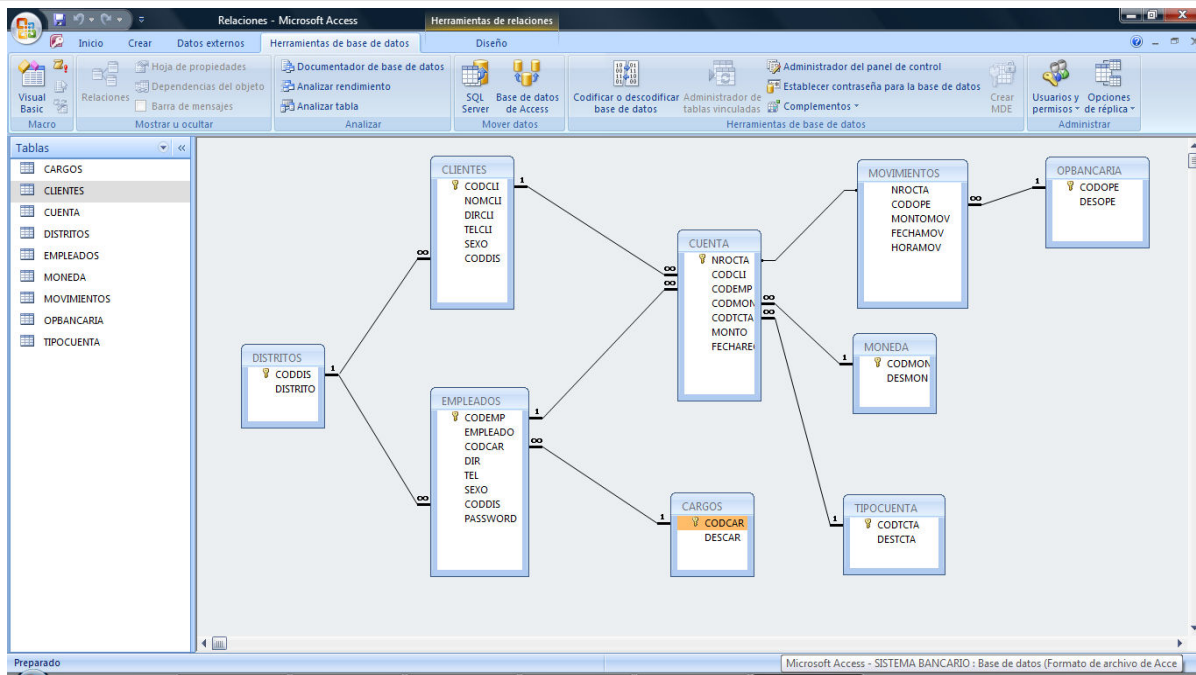


Figura 4.3.8 Relaciones en Access.

Siguiendo éste procedimiento creamos las relaciones que fueron necesarias para nuestro sistema, realizando de antemano un estudio previo del diseño de la base de datos y la asignación de las llaves principales con la finalidad de evitar la duplicidad de datos.

Las consultas son parte de los objetivos de una base de datos que permiten recuperar datos de una tabla, modificarlos e incluso almacenar el resultado en otra tabla.

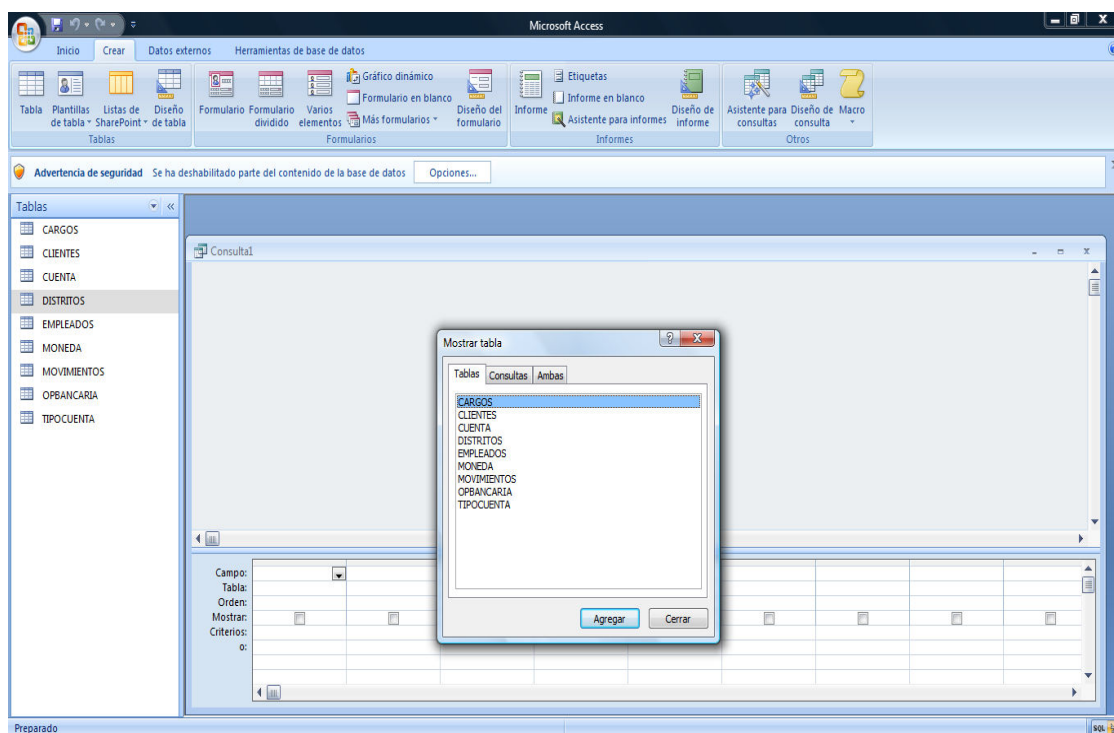
Existen diferentes tipos de consultas:

- Consultas de selección. Son las consultas que extraen o nos muestran datos. Muestran aquellos datos, de una tabla o varias tablas, que cumplen los criterios especificados. Una vez obtenido el resultado podremos consultar los datos para modificarlos (esto se podrá hacer o no según la consulta). Una consulta de selección genera una tabla lógica (se llama lógica porque no está físicamente en

el disco duro sino en la memoria del ordenador y cada vez que se abre se vuelve a calcular).

- Consultas de acción. Son consultas que realizan cambios a los registros. Existen varios tipos de consultas en Access: de acción, de eliminación, de actualización, de datos anexados y de creación de tablas.

Para crear una consulta en la base de datos hacemos click en el botón Diseño de Consulta en la pestaña Crear; al entrar en la Vista Diseño de consulta Access nos pide primero las tablas de las que la consulta sacará los datos con un cuadro de diálogo parecido al mostrado en la figura 4.3.9.



**Figura 4.3.9. Consultas.**



Seleccionamos la tabla de la que queremos extraer datos y hacemos click sobre el botón Agregar. Si queremos obtener datos de varias tablas, agregamos de la misma forma las demás tablas. Aparecerá la ventana Vista Diseño de consultas.

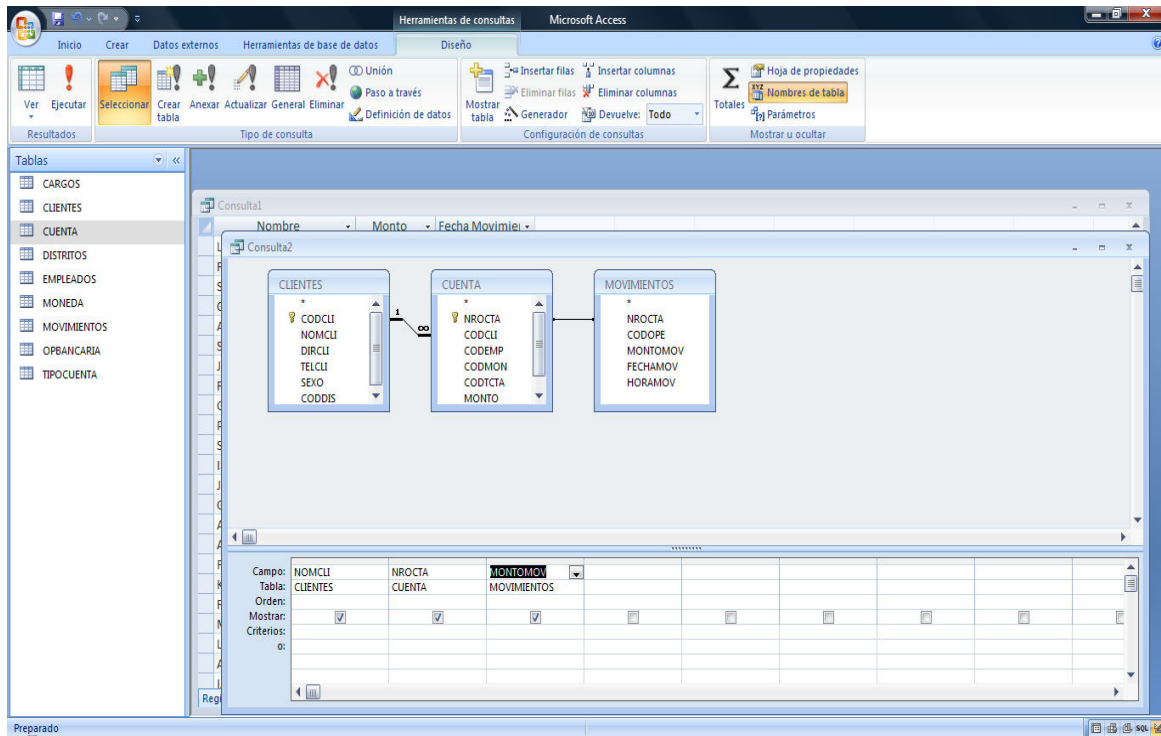
En la parte superior tenemos la zona de tablas, donde aparecen las tablas añadidas con sus correspondientes campos y en la parte inferior denominada cuadrícula QBE definimos la consulta. Cada columna de la cuadrícula QBE corresponde a un campo. Cada fila tiene un propósito que detallamos brevemente a continuación:

- Campo: Seleccionamos el campo a utilizar, que en la mayoría de los casos será el campo a visualizar; puede ser el nombre de un campo de la tabla y también puede ser un campo calculado.
- Tabla: Nombre de la tabla de la que seleccionamos el campo. Es útil para definir consultas basadas en varias tablas.
- Orden: Sirve para ordenar las filas del resultado.
- Mostrar: Si la casilla de verificación aparece desactivada, la columna no aparecerá en el resultado; se suele desactivar cuando queremos utilizar el campo para definir la consulta pero no queremos que aparezca en el resultado.
- Criterios: Sirve para especificar un criterio de búsqueda. Un criterio de búsqueda es una condición que deben cumplir los registros que aparecerán en el resultado de la consulta. Por lo tanto está formado por una condición o varias condiciones unidas por los operadores Y (AND) y O (OR).
- O: Ésta fila y las siguientes se utilizan para combinar condiciones.

Podemos añadir campos a la cuadrícula haciendo doble click sobre el nombre del campo que aparece en la zona de tablas; éste se colocará en la primera columna libre de la cuadrícula. Si queremos que todos los campos de la tabla aparezcan en el resultado de la consulta podemos utilizar el asterisco (\*) cómo sinónimo de todos los



campos. Si se requiere definir campos calculados, éstos son campos obtenidos del resultado de una expresión o de una función. Véase la figura 4.3.10.



**Figura 4.3.10. Definición de la Consulta.**

Para ejecutar una consulta hacemos click sobre el botón Ejecutar de la pestaña Diseño. Véase la figura 4.3.11.

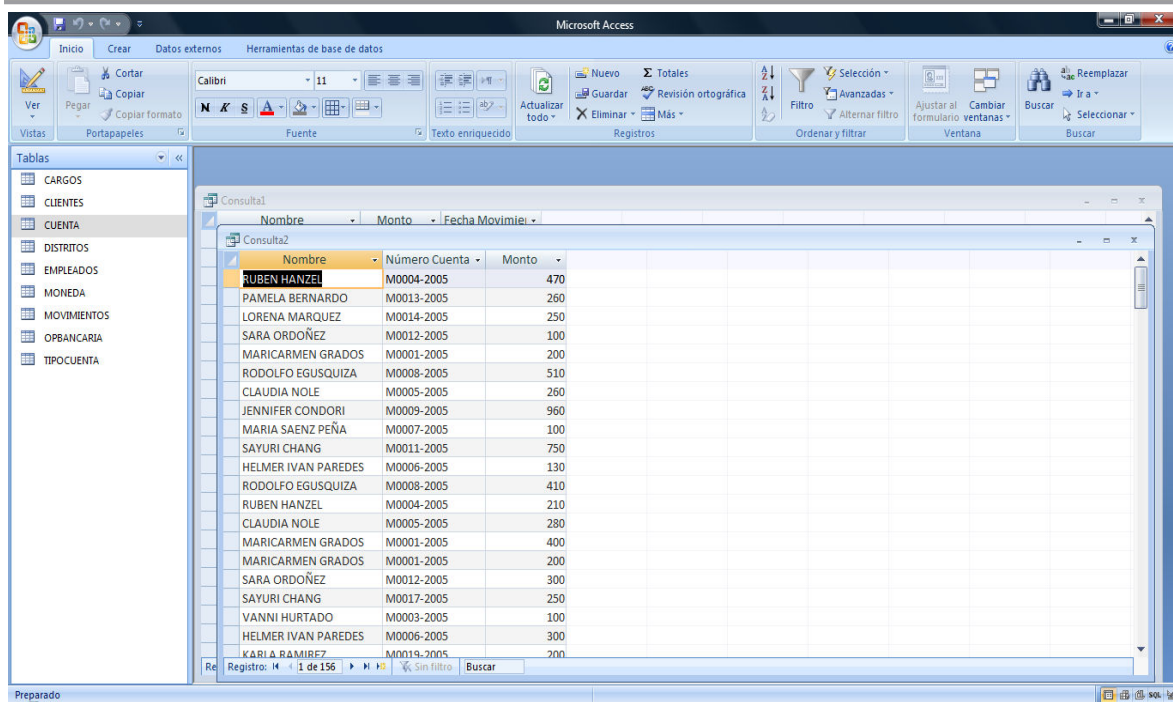


Figura 4.3.11. Ejecutar Consulta.

Quando estamos visualizando el resultado de una consulta, lo que vemos realmente es la parte de la tabla que cumple los criterios especificados, por lo tanto si modificamos algún dato de los que aparecen en la consulta estaremos modificando el dato en la tabla (excepto algunas consultas que no permiten esas modificaciones).

Gracias al empleo de las consultas en nuestro sistema pudimos:

- Elegir campos específicos de tablas específicas,
- Seleccionar información vía criterios,
- Mostrar las informaciones en varios órdenes,
- Obtener datos de varias tablas simultáneamente,
- Calcular totales,
- Crear informes.



En nuestro caso para realizar las consultas utilizamos la opción Diseño de consulta ya que nos permite crear cualquier consulta que necesitemos sin necesidad de ejecutar el asistente. Realizamos las consultas para nuestro sistema siguiendo los pasos antes mencionados y así obtuvimos informaciones específicas contenidas en la base de datos.

Los informes nos sirven para presentar los datos de una tabla o consulta generalmente para imprimirlos. La diferencia básica con los formularios es que los datos que aparecen en el informe sólo se pueden visualizar o imprimir (no se pueden modificar) y en los informes se puede agrupar más fácilmente la información y sacar totales por grupos.

Para crear un informe utilizamos la sección Informes de la pestaña Crear. Diseño de informe abre un informe en blanco en la vista diseño y tenemos que ir incorporando los distintos objetos que queremos aparezca en él. Éste método casi no se suele utilizar ya que en la mayoría de los casos es más cómodo y rápido crear un auto informe o utilizar el asistente y después sobre el informe creado modificar el diseño para ajustar el informe a nuestras necesidades. Primero seleccionamos la tabla o consulta de donde se tomarán los datos y éste será el origen del informe. Si queremos sacar datos de varias tablas lo mejor es crear una consulta para obtener esos datos y luego elegir como origen del informe esa consulta.





## 4.4 DESARROLLO DE LA INTERFAZ DE USUARIO.

La interfaz del sistema, desarrollado en Visual Basic, para operar en ambiente Windows, posee las propiedades de éste y en consecuencia funciona por medio de una serie de ventanas, menús y submenús que facilita la operación del mismo.

Las ventanas son interfaces que permiten al usuario operar la información de la base de datos, en tanto que los menús y submenús son el medio por el que se llega a dichas ventanas.

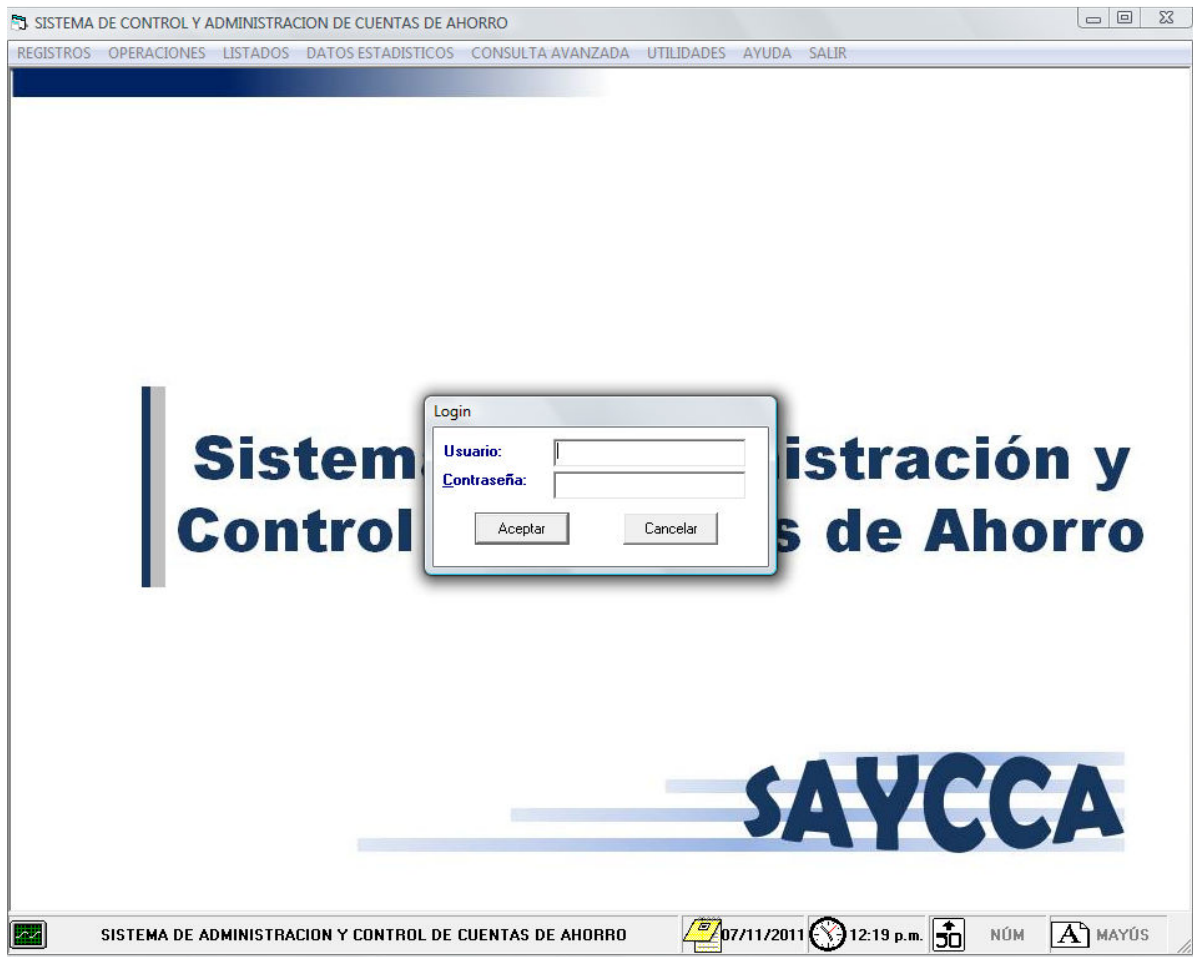
Como parte del sistema se implementó la pantalla de presentación, como lo muestra la Figura 4.4.1, la cual contiene la bienvenida al usuario, el nombre del sistema y un reloj de barra que representa el tiempo de espera para el usuario.



Figura 4.4.1 Pantalla de inicio del Sistema



La siguiente pantalla, figura 4.4.2, muestra el formato de captura del login y password del usuario, teniendo como fondo la ventana principal del sistema. Esta pequeña ventana de captura de datos, es un requisito necesario para dar ingreso al sistema y es parte fundamental de la seguridad del mismo, incluyendo la base de datos.

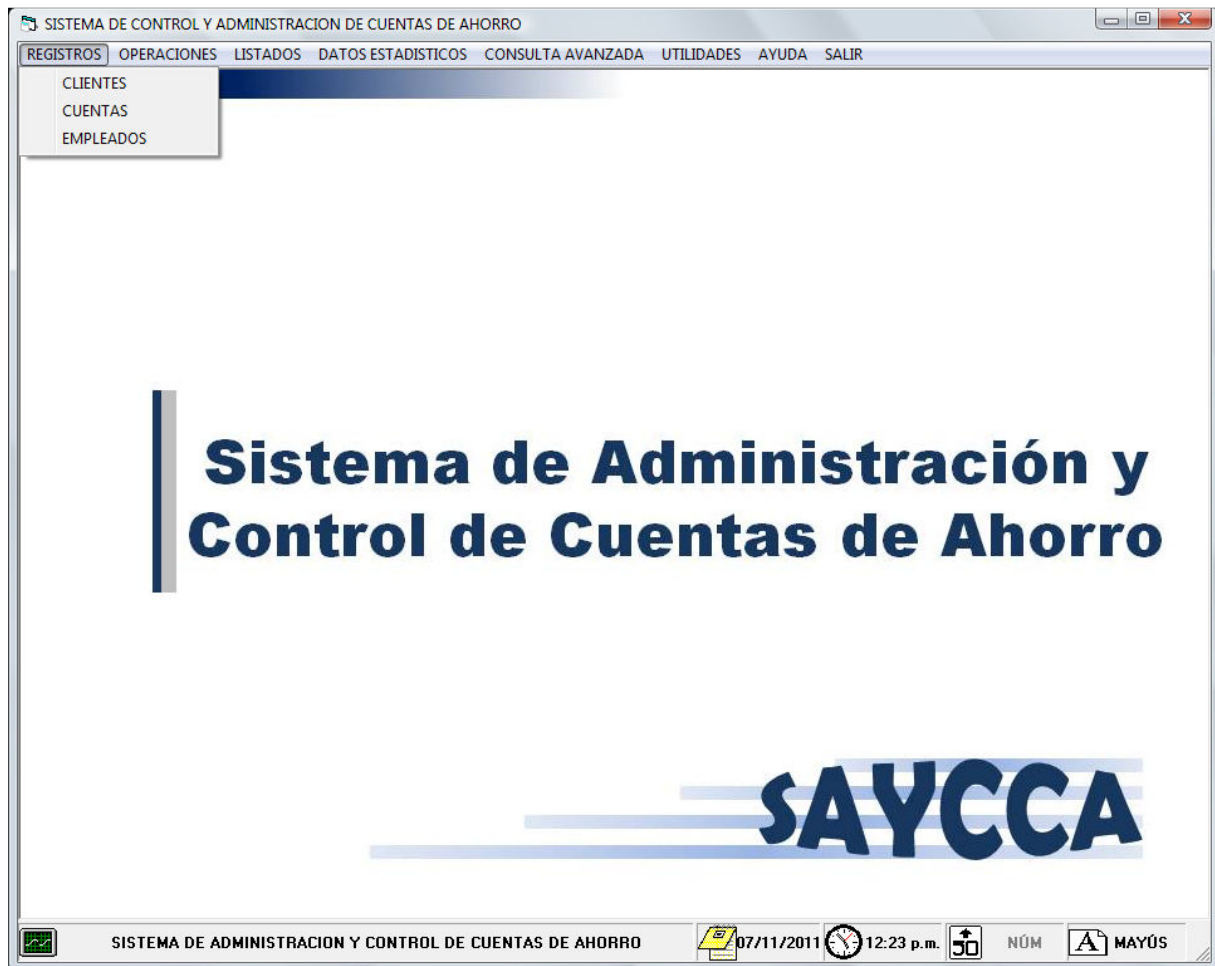


**Figura 4.4.2** Pantalla principal y ventana de acceso de login y password de usuario

La pantalla principal es una ventana con las propiedades de sus similares de Windows, es decir es de tamaño ajustable, cuenta con los tres botones clásicos: minimizar, agrandar y cerrar.



Para darle funcionalidad al sistema, éste se programó con la herramienta propia de Visual Basic la cual permite, como puede verse en la Figura. 4.4.3, implementar una barra de menú con sus respectivos submenús, para desarrollo de sistemas multifuncionales como es el caso del producto desarrollado.



**Figura 4.4.3** Pantalla principal del sistema activa

Otras ventajas funcionales que presenta la pantalla son: la fecha, la hora, así como etiquetas para aviso de las teclas activas de número y mayúsculas, que son de ayuda para el usuario.



Los menús permiten al usuario operar las cuentas de los clientes, a los mismos clientes y a los empleados además de consultar información con opciones adicionales. Estos submenús conducen a las pantallas correspondientes para realizar operaciones sobre los objetos de interés del operador en turno. El acceso a estas y sus funciones dependen de la clave y password, que a su vez dependen del puesto del operador. La Figura 4.4.4 muestra la pantalla de control de clientes.



Fig. 4.4.4. Pantalla de manipulación de datos del cliente.



Para facilitar la programación se aprovecharon las herramientas de cuadro o frame que permiten al programador conjuntar otras herramientas como son cuadros de texto, botones y selectores.

Los cuadros de texto, previamente seguidos por etiquetas, con los nombres correspondientes, e incluidos en el interior del primer cuadro, son necesarios ya que nos permiten capturar y mostrar los datos de la información mínima requerida para el manejo de las cuentas de ahorros, sus propietarios y los empleados.

En el segundo cuadro, dentro del primero, se establecieron dos botones de opciones (BottonOpcion), para elegir el sexo del cliente en turno, considerando que la Sociedad de Ahorro presta el servicio al público en general, sin menoscabo de géneros.

Continuamos con el tercer cuadro, de consulta, y contiene cuatro botones que permiten al operador recorrer la base de datos hasta encontrar al cliente deseado. Los botones en los extremos conducen al principio (Izquierda) o al final (Derecha) de la base de datos. Los del centro permiten un recorrido secuencial de registros, hacia adelante o hacia atrás. Además se cuenta con una subpantalla adicional, donde se pueden realizar búsquedas de clientes de acuerdo a su información.

El último cuadro contiene los botones para realizar las operaciones fundamentales requeridas por la Sociedad de Ahorro como son:

- Agregar clientes, cuentas y empleados
- Guardar, borrar y modificar los registros capturados
- Cancelar la operación
- Salir de la forma activa.

La pantalla de movimientos, permite aplicar retiros y depósitos de efectivo realizados por los cuentahabientes. Esto se hace mediante una serie de acciones secuenciales con la finalidad de que el movimiento se haga de forma correcta.

Para dar inicio a estas operaciones se implementó un botón que activa el acceso a la forma, el cual se encuentra en el lado izquierdo del frame que contiene el textbox de captura de cuenta, como se muestra en la Figura 4.4.5.



La imagen muestra una interfaz de usuario web para el sistema SAYCCA. En la parte superior, hay un menú con opciones: REGISTROS, OPERACIONES, LISTADOS, DATOS ESTADISTICOS, CONSULTA AVANZADA, UTILIDADES, AYUDA y SALIR. El encabezado principal contiene el logo de SAYCCA y el slogan "Sociedad de Ahorro ... tu patrimonio seguro". Debajo, el título "OPERACIONES" precede a un formulario. Este formulario incluye un campo "CUENTA" con tres sub-campos, un campo "FECHA" con el valor "07/11/2011" y un campo "HORA" con "12:26:27 p.m.". Sección "DATOS DEL CLIENTE" con campos para "NOMBRE", "SEXO", "DIRECCIÓN", "ESTADO" y "TELÉFONO". Campos para "SALDO ACTUAL EN" y "NUEVA FECHA DE VENCIMIENTO:". Una pregunta "¿QUE OPERACIÓN DESEA REALIZAR?" está seguida por botones "DEPOSITO" (con una flecha azul hacia la derecha) y "RETIRO" (con una flecha azul hacia la izquierda). Hay también un botón con un icono de documento y otro con un icono de cancelación (una X roja). El pie de página muestra "SISTEMA DE ADMINISTRACION Y CONTROL DE CUENTAS DE AHORRO", la fecha "07/11/2011", la hora "12:26 p.m.", y un teclado virtual con "NÚM" y "MAYÚS".

**Figura 4.4.5. Pantalla para operación de movimiento de cuentas**

Se utilizan “textbox” para mostrar los datos del cliente, condición necesaria para corroborar los datos del cliente al momento de la operación.

La elección de la operación a realizar puede hacerse mediante dos commandbuttons etiquetados con los nombres correspondientes de “retiro” y “deposito”. La selección de cualquiera de estos activa el textbox correspondiente para la captura del monto de la operación. Una vez que se ha escrito la cifra se deberá oprimir “Enter” para continuar la secuencia. Acto seguido, se activará el botón de “registro” que permite guardar el movimiento hecho, así como actualizar la base de datos. El último paso de la secuencia



activa el botón de impresión del recibo del movimiento hecho y que por obligación se debe entregar al cuentahabiente, así como su correspondiente copia para la Sociedad de Ahorro. El recibo aparece en pantalla a fin de que el cajero corrobore los datos y de ser correctos proceda a la impresión. Dicho recibo contiene los datos del movimiento efectuado: nombre del cuentahabiente, número de cuenta, fecha, hora, operación, monto y saldo. La Figura 4.4.6 muestra la imagen del recibo y los botones correspondientes de impresión y de presentación en el block de notas.



**Figura 4.4.6. Recibo de retiro generado por el sistema**

Debido a que la Sociedad de Ahorro requiere revisar los movimientos de las cuentas, así como de quién las efectuó –tanto clientes, como empleados-, las fechas, cantidades y otros datos relacionados con estos, se implementaron varias ventanas que permiten visualizar dicha información. Ésta se muestra de forma horizontal ya que los datos requeridos suelen ser demasiados y en algunos campos las cadenas de información son gran tamaño, como se muestra en la Figura 4.4.7.



CODIGO	EMPLEADO	CARGO	DIRECCION	TELEFONO	SEXO	DELEGACION
EMP0011	Juan V	CAJERO	jaskdaksjd	55555555	M	AZCAPOTZALCO
EMP0012	Is -I grep mysq	CAJERO	abcdelghij	5555555555	M	BENITO JUAREZ
EMP0001	DIANA ANTILLON	GERENTE	AV. LA FONTANA 3650 - LA MOLINA	97085623	F	GUSTAVO A. MADERO
EMP0002	SEBASTIAN DE MOLINA	CAJERO	CALLE LOS NARANJOS 470 DPTD. 305	2-35-6211	M	MILPA ALTA
EMP0003	MARIA LUZ FUENTES	GERENTE	AV. TACNA 2630 - LIMA	5-30-2070	F	ALVARO OBREGON
EMP0004	ERNESTO COLAN	SUBGERENTE	CALLE GRAÑA 1460 MZ. T - MIRAFLORES	4-15-9314	M	MAGDALENA CONTRERAS
EMP0005	JESUS ROJAS	CAJERO	CALLE LA ALAMEDA 1950 ALT. CUADRA 20	97025631	M	XOCHIMILCO
EMP0006	FERMIN DE LA CRUZ	CONTADOR Y/O ANALIS	CALLE ORBEGOSO 270 ESQ. CUADRA 8	3-25-6282	M	XOCHIMILCO
EMP0007	OMAR OLIVIERI	GERENTE	CALLE TREMI 260 ALT. CUADRA 6	97025631	M	MIGUEL HIDALGO
EMP0008	BERTHA MARTINEZ	SUBGERENTE	NIQUEL 204. COL. GUADALUPE INN	3231-1213	F	IZTAPALAPA
EMP0009	JOEL GARCÍA	CONTADOR Y/O ANALIS	KINCHIL 14 COL. HEROES DE PADIERNA	51235121	M	IZTAPALAPA
EMP0010	RODOLFO MORALES	ADMINISTRADOR DBA	UNIVERSIDAD 1311 COL. DEL CARMEN	56512323	M	IZTAPALAPA

Figura 4.4.7 Pantalla de consulta de empleados.

Como se observa, cuenta con dos botones de operación: el de impresión y el de salida del módulo. El primero, permite imprimir la información que aparece en pantalla, con opción de edición en el block de notas, como se muestra en la Figura 4.4.8. Esta facilita el manejo de la información para la creación de reportes, notas y oficios que son necesarias en las actividades cotidianas en la institución. El botón de salida, cuando se activa, permite salir de este módulo.





CODIGO	NOMBRE	CARGO	DIRECCION	TELEFONO
EMP0011	Juan V	CAJERO	jaskdaksjd	55555555
EMP0012	Is -l   grep mysql	CAJERO	abcdefghijkl	55555555
EMP0001	DIANA ANTILLON	GERENTE	AV. LA FONTANA 3650 - LA MOLINA	97085623
EMP0002	SEBASTIAN DE MOLINA	CAJERO	CALLE LOS NARANJOS 470 DPTO. 305	2-35-6211
EMP0003	MARIA LUZ FUENTES	GERENTE	AV. TACNA 2630 - LIMA	5-30-2070

**Figura 4.4.8. Reporte de empleados generado por el sistema**

La información estadística necesaria para la institución -para planeación, análisis y toma de decisiones- puede consultarse por medio de la opción “Datos Estadísticos”, ofrece en las opciones de submenú acceso a consulta de movimientos, movimientos por mes, resumen de cuenta, etc.

Para facilitar el manejo de la información consultada se agregó el botón de “Exportar a Excel”, la cual permite trasladar los datos en pantalla a un archivo en formato de la hoja de cálculo mencionada. Esta facilidad ayudará a los usuarios a manipular eficientemente los datos obtenidos, facilitando su disponibilidad, mejorando su



presentación visual, desarrollar gráficos que clarifiquen la idea que se quiere transmitir.

La Figura 4.4.9 muestra un ejemplo de la ventana de “Datos Estadísticos”.

CODIGO	NOMBRE	DIRECCION	TELEFONO	SEXO	DELEGACION
C0003-2005	VANNI HURTADO	JR. QUILCA 148-LIMA	2-85-4414	F	ALVARO OBREGON
C0039-2011	JAVIER INIESTA GONZALEZ	EQEWVWV DSDSA ASAD	77979862	M	ALVARO OBREGON
C0040-2011					ALVARO OBREGON
C0041-2011	CARLOS RAMIREZ	EQEQWEQ 121	13123 13123		ALVARO OBREGON
C0043-2011	JOHN LOPEZ	IUYIUYDIYIUY IUYDIY12323	2122102121	M	ALVARO OBREGON
C0004-2005	PABLO SALAZAR	URB. FELIPE ALVA, CALLE 12 MZ. J-VILLA VICTORIA-	4-70-1230	M	AZCAPOTZALCO
C0006-2005	HELMER IVAN PAREDES	CALLE BELISARIO GUTIERREZ 204-VILLA VICTORIA	4-36-0963	M	AZCAPOTZALCO
C0007-2005	CLAUDIA NOLE	JR. LUNA PIZARO 102-VILLA VICTORIA	7-95-9489	F	AZCAPOTZALCO
C0008-2005	JENNIFER CONDORI	PJ. VILLEGAS 124-SURQUILLO	3-66-8656	F	AZCAPOTZALCO
C0009-2005	DANIEL CALDERON	AV. TOMAS MARSANO 1890	2-24-9603	M	AZCAPOTZALCO
C0010-2005	KIARA GARCIA	AV. NICOLAS DE PIEROLA 751-SURQUILLO	5-74-2662	F	AZCAPOTZALCO
C0038-2011	JAVIER HERNANDEZ MARTINEZ	HKJHKJKHJKL	2525253232	M	AZCAPOTZALCO
C0042-2011	ROBERTO PEREA	QEWOIQUWOIQ QwWwQ 121	13123 13123	M	AZCAPOTZALCO
C0001-2005	DUO MAXWELL	CIUDAD CAPITAL 140 - LIMA	2-10-9480	M	BENITO JUAREZ
C0012-2005	RUBEN HANZEL	CALLE ANTUNEZ MAYOLO 194-MONTEERRICO	4-52-3415	M	BENITO JUAREZ
C0030-2005	RODOLFO MORALES	AV. CONGRESO 1550, COL. ROMA	566644444	M	COYDACAN
C0036-2011	RODRIGO GONZALEZ		5656-9811	M	COYDACAN
C0013-2005	SARA ORDOÑEZ	AV. LOS CIRUELOS 830 DPTO. 302	4-15-3206	F	CUAJIMALPA
C0014-2005	GINO CORTEZ	CALLE ORDOÑEZ 209 DPTO. 140	98988144	M	CJAUHTEMOC
C0028-2005	FABIOLA DE LA COTERA	CALLE SOLER 1360 BLOQUE "F", DPTO. 580	97004703	F	CJAUHTEMOC
C0034-2011	IAN ANDERSON		6558-8989	M	CJAUHTEMOC
C0015-2005	ANDREA SOTELO	AV. LA MOLINA 1730	4-56-8974	F	GUSTAVO A. MADERO
C0016-2005	SAYURI CHANG	AV. LA FONTANA 215-LA MOLINA	64578123	F	GUSTAVO A. MADERO
C0017-2005	JUAN CARLOS LLOSA	RESIDENCIAL LA MOLINA 1260 DPTO. 220	69845120	M	GUSTAVO A. MADERO
C0019-2005	RODOLFO EGUSQUIZA	PJ. REAL 109-BELLAVISTA	2-81-0207	M	IZTAPALAPA
C0032-2005	LUIS MOYA ZEPEDA	JHKHKJKHJH	51280-102	M	IZTAPALAPA

Figura 4.4.9 Reporte de datos estadísticos de clientes.

El sistema cuenta con la opción de utilidades que permite a los usuarios invocar herramientas como calculadora, block de notas, Word y Excel, sin necesidad de salir del ambiente del sistema.



## **4.5 INTEGRACIÓN, PRUEBAS y MANTENIMIENTO.**

### **4.5.1 Pruebas.**

Las pruebas constituyen una parte integral y vital del desarrollo del sistema. Se realizan con el propósito de descubrir defectos y se establecen para mejorar la calidad del sistema antes de que sea liberado.

En la realización de pruebas es básico que estas se lleven a cabo por personas distintas a los desarrolladores del sistema, tanto para evitar una simple verificación de que el sistema funcione correctamente, como para probar que ese sistema ha sido concebido e interpretado correctamente.

Los casos de prueba deben ser escritos tanto para condiciones de entrada validas o inesperada, como para condiciones validas y esperadas.

Un principio deducido de la experiencia y observación de pruebas de diferentes sistemas, es que la probabilidad de encontrar errores adicionales en una sección del sistema, es proporcional al número de errores ya encontrados en la misma sección.

La fase de pruebas da inicio luego de que las diferentes unidades de diseño han sido desarrolladas y probadas por separado. Durante su desarrollo, el sistema se emplea de forma experimental para asegurar que el software no falle, es decir que funcione de acuerdo a sus especificaciones y a la manera que los usuarios esperan que lo haga, y de esta forma poder detectar cualquier anomalía antes de que el sistema sea puesto en producción.



Para evaluar el desenvolvimiento del sistema, en esta fase se llevan a cabo varios niveles de prueba que nos proporcionan una oportunidad para eliminar cualquier problema que se presente:

- Pruebas de caja negra
- Pruebas de caja blanca
- Pruebas de integración
- Pruebas de rendimiento
- Pruebas de regresión
- Pruebas de volumen
- Pruebas de seguridad
- Pruebas de aceptación
- Pruebas alfa
- Pruebas beta

Cuando se considera que un módulo está terminado se realizan las pruebas sistemáticas, el objetivo de estas es buscar fallos a través de un criterio específico, estos criterios se denominan “pruebas de caja negra y de caja blanca”.

#### **4.5.1.1 Pruebas de Caja Negra (Funcional).**

Esta prueba se realiza con base a los requerimientos sin conocimiento sobre cómo fue construido el sistema y usualmente dirigidas a los datos, es decir se enfocan directamente en el exterior del módulo o componente, sin importar el código, son pruebas funcionales en las que se trata de encontrar fallas en las que no se atiende a su especificación, ya que éste es una caja negra y su comportamiento solo puede ser determinado al estudiar sus entradas y las salidas relacionadas a éstas. Las pruebas son un enfoque complementario a las pruebas de caja blanca que intentan reconocer errores con las siguientes características:

---

---



- Funciones incorrectas o ausentes
- Errores de interfaz
- Errores de estructura de datos o en accesos a bases de datos externas
- Errores de rendimiento
- Errores de inicio y terminación

#### **4.5.1.2 Pruebas de Caja Blanca (Estructural).**

Este tipo de pruebas son el contraste al de la caja negra, debido a que quién realiza la prueba puede analizar el código, estas pruebas son mucho mas amplias, normalmente se denominan pruebas de cobertura o pruebas de caja transparente, al total de pruebas de caja blanca se le llama cobertura, la cobertura es un número porcentual que indica cuanto código del programa se ha probado. Este tipo de prueba se realiza cuando se desarrolla un sistema, debido a que el programador conoce la lógica del sistema y la estructura de un componente para poder obtener los datos de prueba, 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 del sistema en general, buscando:

- Garantizar que todas las trayectorias independientes dentro de un módulo hayan sido ejecutadas dentro de éste al menos una vez.
- Ejecutar todos los lados de las decisiones lógicas (verdaderas y falsas).
- Ejecutar todos los ciclos o bucles en sus límites operacionales.
- Ejecutar las estructuras de datos internas para asegurar su validez.

#### **4.5.1.3 Pruebas de Integración.**

Estas pruebas se realizan a un grupo de programas para asegurar que los datos u controles sean pasados adecuadamente entre ellos mismos. 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. Los tipos fundamentales de integración son:

- **Integración incremental.** Se combina el siguiente módulo que se debe probar con el conjunto de módulos que ya han sido probados.
- **Integración no incremental.** Se prueba cada módulo por separado y luego se integran todos de una vez y se prueba el sistema completo.
- **Ascendente.** Se comienza por los módulos hoja o base.
- **Descendente.** Se comienza por el módulo raíz y se va expandiendo hacia abajo.

El objetivo es tomar los módulos probados en unidad y construir una estructura que esté de acuerdo con lo que dicta el diseño.

La integración contempla aspectos como:

- **Integridad Semántica:** Cuando existen validación a través de programas o de un manejador de bases de datos.
- **Integridad de Entidades:** No se deben de aceptar campos nulos en la llave primaria.
- **Integridad de Referencia:** Las llaves foráneas corresponden a la llave primaria.

#### 4.5.1.4 Pruebas de Regresión.

Cada vez que se añade un módulo nuevo como parte de las pruebas de integración, el software cambia: se establecen nuevos caminos en el flujo de datos, pueden existir nuevas I/O, y se invoca a una nueva lógica de control, lo cual puede ocasionar problemas con funciones que ya trabajaban correctamente. Las pruebas de regresión consisten en volver a ejecutar un subconjunto de pruebas que se han llevado a cabo anteriormente, para asegurarse que los cambios no han ocasionado efectos colaterales indeseados.



Por otra parte todos los sistemas sufren una evolución a lo largo de la vida activa. En cada nueva versión se corrigen defectos o se añaden nuevas funciones, o ambas cosas. En cualquier caso, una nueva versión exige una nueva verificación. Si éstas se han sistematizado en una fase anterior, ahora pueden volver a realizarse automáticamente, simplemente para comprobar que las modificaciones no provocaron errores donde antes no los había.

Por consiguiente, ésta prueba detectará fallas en algún módulo debido a modificaciones realizadas con anterioridad.

#### **4.5.1.5 Pruebas de Volumen.**

Estas pruebas se realizan para verificar el funcionamiento adecuado y eficiente de una aplicación bajo condiciones extremas de operación, como jornadas diarias y normales de procesos diarios y continuos, comprobando la inexistencia del mal funcionamiento del sistema.

#### **4.5.1.6 Pruebas de Seguridad.**

Este tipo de prueba intenta asegurar que los mecanismos de protección del sistema sean adecuados y evitar la entrada inválida. Durante esta prueba el encargado de la prueba funge como un individuo ajeno que desea vulnerar la seguridad.

#### **4.5.1.7 Pruebas de Aceptación.**

Estas pruebas las realiza el cliente. Son básicamente pruebas funcionales sobre el sistema completo y buscan una cobertura de la especificación de requisitos. Estas pruebas no se realizan durante el desarrollo, ya que sería impresentable de cara al



cliente, sino una vez realizada todas las pruebas de integración por parte del desarrollador.

La experiencia muestra que aún después del más cuidadoso proceso de pruebas por parte del desarrollador, quedan una serie de errores que solo aparecen cuando el cliente se pone a usarlo.

Por esta razón, muchos desarrolladores ejercitan unas técnicas denominadas “Pruebas Alfa y Beta”, las cuales serán descritas a continuación.

#### **4.5.1.8 Pruebas Alfa y Beta.**

Las pruebas alfa consisten en invitar al usuario final al entorno de desarrollo del sistema. Se trabaja en un entorno controlado y el usuario siempre tiene un experto a la mano para ayudarlo a usar el sistema y para analizar los resultados.

Las pruebas beta vienen después de las pruebas alfa y se desarrollan en el entorno del usuario final, un entorno que esta fuera de control. Aquí el usuario se queda a solas con el sistema y trata de encontrar fallas que posteriormente informará al desarrollador.

Las pruebas alfa y beta son habituales en productos que se van a vender a muchos clientes. Algunos de los potenciales compradores se prestan a estas pruebas bien por ir entrenando a su personal con tiempo, bien a cambio de alguna ventaja económica (mejor precio sobre el producto final, derecho a mantenimiento gratuito, a nuevas versiones, etc.).

La fase de pruebas absorbe una buena porción de los costos de desarrollo de software. Además, se muestra renuente a un tratamiento matemático o simplemente automatizado. Su ejecución se basa en la metodología que se van desarrollando con la experiencia.

---

---





Aunque se han desarrollado miles de herramientas de soporte de ésta fase, todas han limitado su éxito a entornos muy concretos, frecuentemente sólo sirviendo para el producto. Sólo herramientas muy generales como analizadores de complejidad, sistemas de ejecución simbólica y mediadores de cobertura han mostrado su utilidad en un marco más amplio.

#### **4.5.2 Mantenimiento del sistema.**

En este apartado describiremos los tipos de mantenimientos aplicables al Sistema.

- Preventivo
- Correctivo
- Perfectivo
- Adaptativo
  - Aumentativo
  - Tecnológico

##### **4.5.2.1 Mantenimiento Preventivo.**

Es la actividad en la cual se realizan cambios a la aplicación para mejorar el mantenimiento futuro, la estabilidad y confiabilidad en la operación. También es útil para proporcionar bases seguras sobre las que podrán implementarse mejoras posteriores.

##### **4.5.2.2 Mantenimiento Correctivo.**

La primera actividad del mantenimiento se da ocasionalmente cuando la prueba del software no haya descubierto todos los errores latentes de un sistema. Durante el uso del sistema se encontrarán errores, los cuales deber ser informados al equipo de

---

---



desarrollo. El proceso que incluye el diagnóstico y corrección de uno o más errores se denomina mantenimiento correctivo.

#### **4.5.2.3 Mantenimiento Perfectivo.**

Se realiza cuando existe la necesidad de optimización de procesos, sin que cambien forzosamente los requerimientos. La especificación permite entender claramente el impacto de los cambios de manera que estos se implanten confiadamente.

#### **4.5.2.4 Mantenimiento Adaptativo.**

El mantenimiento adaptativo se debe a cambios en el ambiente del programa y a la adaptación de nuevas unidades o módulos. En estudios realizados se ha observado que alrededor de 65% del mantenimiento era perfectivo, el 18% adaptativo y el 17% correctivo.

En este tipo de mantenimiento se encuentran implícitos el aumentativo y el tecnológico. La vida útil estimada del software de aplicación puede ser muy extensa, pero considerando la evolución del ambiente, en la práctica éste puede volverse obsoleto.

Por lo tanto, el mantenimiento adaptativo es una actividad que modifica al software para que las interacciones sean adecuadas al entorno cambiante.

#### **4.5.2.5 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.



#### **4.5.2.6 Mantenimiento Tecnológico.**

Esta actividad que contribuye al mantenimiento se da debido a todo cambio importante en la informática. Si en un periodo de tiempo surgen nuevas generaciones de hardware, regularmente aparecen nuevos sistemas operativos o nuevas versiones de los antiguos, y frecuentemente se mejoran o modifican los equipos periféricos y otros elementos de sistemas.

La mayor parte del mantenimiento de adaptaciones se hace como respuesta a la aparición de nuevos problemas de la empresa, nuevas necesidades de información o nuevas ideas de mejoras. Estas actividades reciben el nombre de mejoras del sistema.

El objetivo de las mejoras al sistema es modificar o ampliar el sistema de aplicaciones como respuesta a las necesidades cambiantes de la empresa. Un sistema es propenso a fallar eventualmente, por lo que como parte del mantenimiento, se recomienda que se realicen respaldos de la información contenida en la base, esto previendo que el sistema llegara a tener alguna falla grave, o por algún imprevisto, sea borrado total o parcialmente del servidor en donde se aloja, así la información por lo menos se encontraría en resguardo.

#### **4.6 Generación de Reportes**

En la industria es bien sabido la importancia de monitorear el comportamiento de la empresa, a los Ejecutivos, Gerentes y Directores les interesa conocer los Reportes Mensuales para evaluar situaciones como:

- Toma de Decisiones para Medición de Riesgos.
- Desarrollo de Modelos de Evaluación de Costos.
- Áreas de Oportunidad.
- Procesos como el Benchmarking.



- Mejoras en la Productividad.

Dentro de nuestro sistema también existe la posibilidad de generar reportes, el sistema, como se había mencionado con anterioridad, cuenta con un módulo de reportes (Módulo de Estadísticas, capítulo 3.4.3), dentro de este módulo existen las siguientes opciones de Generación de Reportes:

- Reporte de Listado de Clientes.
- Reporte de Listado de Empleados.
- Reporte de Listado de Cuentas.
- Reporte de Datos Estadísticos de Movimientos.
- Reporte de Datos Estadísticos de Movimientos por Mes.
- Reporte de Datos Estadísticos de Resumen por Tipo de Cuenta.
- Reporte de Datos Estadísticos de Resumen por Estados.
- Reporte de Datos Estadísticos de Cuentas por Meses.

A continuación explicaremos cada uno de ellos y se mostrara el Reporte que genera el Sistema:

#### 4.6.1 Reporte de Listado de Clientes.

Es un compendio de todos los clientes que se encuentran alojados en la Base de Datos, consta de cuatro campos los cuales son: Código, Nombre, Dirección, Teléfono, Sexo y Distrito. Este reporte es usado para contabilizar cuantos clientes tiene la institución, los datos proporcionados, informan de que tipo de clientes son, su lugar de residencia y su número telefónico como se puede observar en la tabla 4.6.1.

**Acceso:** Contador/Analista, Gerente y Subgerente.

**Generado:** Mensualmente los primeros días hábiles.

Sistema de Administración y Control de Cuentas de Ahorro					
Reporte de Listado de Clientes					
Código	Nombre	Dirección	Teléfono	Sexo	Distrito
C0001-2005	DUO MAXWELL	CIUDAD CAPITAL 140	552768815	M	D003



C0002-2005	MARICARMEN GRADOS	AV. 28 DE JULIO 1560	555840507	F	D001
C0003-2005	VANNI HURTADO	JR. QUILCA 148	552854414	F	D001
C0004-2005	PABLO SALAZAR	URB. FELIPE ALVA, CALLE 12 MZ. J -VICTORIA	554701230	M	D002
C0005-2005	MARIA SAENZ PEÑA	AV. TOMAS MARSANO 2670	555447248	F	D002
C0006-2005	HELMER IVAN PAREDES	CALLE BELISARIO GUTIERREZ 204-VICTORIA	554360963	M	D002
C0007-2005	CLAUDIA NOLE	JR. LUNA PIZARO 102-VILLA VICTORIA	557959489	F	D002
C0008-2005	JENNIFER CONDORI	PJ. VILLEGAS 124-SURQUILLO	553668656	F	D002
C0009-2005	DANIEL CALDERON	AV. TOMAS MARSANO 1890	552249603	M	D002
C0021-2005	PAMELA BERNARDO	AV. BARRENECHEA 480-SAN ISIDRO	552607430	F	D011
C0022-2005	SYDNA PERTUSA	JR. ALBARRA 4890-SAN JUAN MIRAFLORES	5578465208	F	D012

**Tabla 4.6.1 Reporte de Listado de Clientes.**

#### 4.6.2 Reporte de Listado de Empleados.

De la misma forma que el reporte anterior, esta información cuenta con los mismos campos, solo que este es para el registro de empleados lo podemos visualizar en la tabla 4.6.2, usado básicamente para control de ellos: asistencia, pagos por nómina, cargo y áreas en donde se encuentran involucrados.

**Acceso:** Contador/Analista, Gerente y Subgerente.

**Generado:** Mensualmente los primeros días hábiles.

Sistema de Administración y Control de Cuentas de Ahorro					
Reporte de Listado de Empleados					
Código	Nombre	Dirección	Teléfono	Sexo	Distrito
EMP0001	DIANA ANTILLON	AV. LA FONTANA 3650 - LA MOLINA	5597085623	F	D007
EMP0002	SEBASTIAN DE MOLINA	CALLE LOS NARANJOS 470 DPTO. 305	5523562118	M	D012
EMP0003	MARIA LUZ FUENTES	AV. TACNA 2630	5553020703	F	D001
EMP0004	ERNESTO COLAN	CALLE GRAÑA 1460 MZ. T - MIRAFLORES	5541593145	M	D010
EMP0005	JESUS ROJAS	CALLE LA ALAMEDA 1850 ALT. CUADRA 20	5597025631	M	D020
EMP0006	FERMIN DE LA CRUZ	CALLE ORBEGOSO 270 ESQ. CUADRA 8	5532562822	M	D025
EMP0007	OMAR OLIVIERI	CALLE TREMI 260 ALT. CUADRA 6	5597025631	M	D011

**Tabla 4.6.2 Reporte de Listado de Empleados.**

#### 4.6.3 Reporte de Listado de Cuentas



Esta tabla tiene siete campos, Número de Cuenta, Nombre de Cliente, Nombre del Empleado, Tipo de Moneda con que se depósito, Tipo de Cuenta, el monto del dinero que se depósito y la fecha cuando se realizó.

Este reporte es uno de los más importantes usado por el Gerente, el subgerente y el Contador, es parte primordial de los procesos de la Institución, ya que de una manera generalizada nos presenta las cuentas bancarias que están registradas en la Institución. El campo número de cuenta del cliente, es un folio que se genera para identificar al cliente, posteriormente el empleado que registro la cuenta, para aclaración alguna, después con que moneda fue realizada la transacción, el tipo de cuenta, las cuales son ahorro normal, a plazos, cuenta corriente y ahorro fácil, este campo también nos informa que sistema es el que prefiere o se inclinan más los clientes, información relevante para la toma de decisiones, ya que se puede optar por aperturar otras variantes de este sistema o crear otros similares, todo esto para incrementar el número de clientes, después tenemos el monto, que es la cantidad disponible del cliente y finalmente la fecha en que se apertura la cuenta, lo podemos observar en la tabla 4.6.3.

**Acceso:** Contador/Analista, Gerente y Subgerente.

**Generado:** Semanalmente Viernes por la tarde.

Sistema de Administración y Control de Cuentas de Ahorro Reporte de Listado de Cuentas Bancarias						
Cuenta	Cliente	Empleado	Moneda	Tipo Cuenta	Monto	Fecha
M0029-2011	CARMEN MARITZA MORALES SAAVEDRA	SEBASTIAN DE MOLINA	DOLARES	AHORRO FACIL	\$12,000.00	08/04/2011
M0001-2005	MARICARMEN GRADOS	DIANA ANTILLON	PESOS	AHORRO NORMAL	\$2,800.00	11/09/2005
M0002-2005	DUO MAXWELL	SEBASTIAN DE MOLINA	DOLARES	AHORRO NORMAL	\$1,800.00	11/11/2005
M0009-2005	JENNIFER CONDORI	MARIA LUZ FUENTES	PESOS	CUENTA CORRIENTE	\$1,300.00	16/10/2005
M0018-2005	FABIOLA DE LA COTERA	FERMIN DE LA CRUZ	PESOS	A PLAZOS	\$1,100.00	18/01/2006

**Tabla 4.6.3 Reporte de Listado de Cuentas Bancarias.**



#### 4.6.4 Reporte de Datos Estadísticos de Movimientos.

Información que debe estar disponible diariamente, es un monitoreo del comportamiento que se está generando de acuerdo a como los clientes están manejando sus cuentas, si están depositando o están haciendo retiros, la fecha en que se están realizando y la hora de mas afluencia, este reporte nos indica si existen más depósitos o mas retiros de dinero, en que fechas los están realizando, y cuál es la cantidad que se está moviendo, a la Institución es de vital importancia esta información, pues si existen más retiros, poner cartas en el asunto y detectar el porqué de estos retiros, si son depósitos, que cantidad están depositando y cada cuando, si es en días de pago(quincena) o a mediados de esta, esto para manejarlo con un poco mas de mercadotecnia.

También este es un reporte muy usado por el Gerente, ya que es una información global del movimiento en la empresa y en la respuesta a resultados, esto lo podemos visualizar mejor en la tabla 4.6.4.

**Acceso:** Contador/Analista, Gerente y Subgerente.

**Generado:** Diariamente en el cierre del Sistema.

<b>Sistema de Administración y Control de Cuentas de Ahorro</b>				
<b>Reporte de Datos Estadísticos de Movimientos.</b>				
<b>Número de Cuenta</b>	<b>Operación</b>	<b>Monto</b>	<b>Fecha</b>	<b>Hora</b>
M0004-2005	DEPOSITO	\$470.00	19/07/2005	07:50:00 p.m.
M0013-2005	DEPOSITO	\$260.00	28/08/2005	01:01:00 p.m.
M0012-2005	DEPOSITO	\$100.00	15/10/2005	04:29:00 p.m.
M0012-2005	DEPOSITO	\$300.00	01/07/2006	08:40:23 p.m.
M0017-2005	DEPOSITO	\$250.00	01/09/2006	04:25:18 p.m.
M0023-2011	DEPOSITO	\$500.00	04/05/2011	12:08:42 a.m.
M0024-2011	DEPOSITO	\$50.00	04/05/2011	12:18:32 a.m.
M0027-2011	DEPOSITO	\$100.00	05/07/2011	06:36:03 p.m.
M0027-2011	DEPOSITO	\$100.00	05/07/2011	06:45:24 p.m.

**Tabla 4.6.4 Reporte de Datos Estadísticos de Movimientos.**



#### 4.6.4.1 Reporte de Datos Estadísticos de Movimientos por Mes.

Este reporte es muy similar al anterior mostrado en la tabla 4.6.4.1, solo que tiene la variante que es generado de acuerdo al mes, disponible en la pantalla del sistema con botones del mes que se desea consultar, podemos generarlo de manera automática y nos da en una forma rápida el detalle de los movimientos que se han generado de acuerdo al mes que se desea saber, es una rápida visualización para identificar alguna aclaración, ya que se le da seguimiento inmediatamente localizando el mes en que se realizó la transacción y nos da el detalle de todo lo que se realizó en ese mes.

**Acceso:** Contador/Analista, Gerente y Subgerente.

**Generado:** En el momento que se desee.



Sistema de Administración y Control de Cuentas de Ahorro				
Reporte de Datos Estadísticos de Movimientos por Mes (Julio)				
Numero de Cuenta	Operación	Monto	Fecha	Hora
M0004-2005	DEPOSITO	\$470.00	19/07/2005	07:50:00 p.m.
M0006-2005	DEPOSITO	\$130.00	07/07/2005	10:25:00 a.m.
M0005-2005	DEPOSITO	\$280.00	01/07/2005	04:49:35 p.m.
M0001-2005	DEPOSITO	\$400.00	01/07/2006	08:34:23 p.m.
M0027-2011	DEPOSITO	\$100.00	05/07/2011	03:22:52 p.m.
M0027-2011	DEPOSITO	\$100.00	05/07/2011	06:31:55 p.m.
M0027-2011	RETIRO	\$100.00	05/07/2011	10:57:29 p.m.
M0027-2011	RETIRO	\$100.00	05/07/2011	11:03:03 p.m.
M0027-2011	RETIRO	\$100.00	05/07/2011	11:15:19 p.m.
M0027-2011	RETIRO	\$100.00	05/07/2011	11:24:29 p.m.

#### 4.6.4.1 Reporte de Datos Estadísticos de Movimientos por mes.





#### 4.6.4.2 Reporte de Datos Estadísticos de Resumen por Tipo de Cuenta.

Este reporte también es generado en pantalla, con dos botones, el primero llamado Clientes por tipo de Cuenta y el segundo Ingresos por tipo de Cuenta, es un resumen de los clientes que nos indica con qué tipo de cuenta están registrados y el monto que está almacenado en ese tipo de cuenta. Este reporte es usado para tener un conocimiento global de lo registrado en la Institución, los ejecutivos lo generan en el momento que lo deseen y les da un conocimiento general sin entrar a detalle, cualquier anomalía o bajo rendimiento, puede ser detectado por ellos y lo canalizan al área correspondiente para su aclaración lo podemos ver en la tabla 4.6.4.2.

**Acceso:** Contador/Analista, Gerente y Subgerente.

**Generado:** En el momento que se desee.

Pesos

Dólares

Euros

**CLIENTES POR TIPO DE CUENTA**

**INGRESOS POR TIPO DE CUENTA**

Tipo de Cuenta	Total de Clientes
AHORRO	6
MONEDERO	8
PLAZO A 30 DIAS	9
PLAZO A 60 DIAS	7
PLAZO A 90 DIAS	6

Tipo de Cuenta	Total de Ingresos
AHORRO	\$12,470.00
MONEDERO	\$19,310.00
PLAZO A 30 DIAS	\$31,290.00
PLAZO A 60 DIAS	\$67,333.30
PLAZO A 90 DIAS	\$20,470.00

#### RESUMEN

TOTAL DE CLIENTES CON CUENTA EN LA SOCIEDAD DE AHORRO	<b>29</b>
TOTAL DE INGRESOS POR CUENTAS	<b>\$83,540.00</b>
<b>Resumen</b>	

**Tabla 4.6.4.2 Reporte de Datos Estadísticos de Resumen por Tipo de Cuenta.**



#### 4.6.4.3 Reporte de Datos Estadísticos de Resumen por Delegación.

Cuenta con tres campos: estado o distrito, total de clientes y sus ingresos, también de la misma forma que los anteriores, es generado en pantalla y solo son estadísticos, es decir, la información se presenta de una manera resumida, en el caso de este reporte, la información va canalizada a las delegaciones, un resumen de el número de los clientes que tiene cada estado y el monto o ingreso.

Para generación de este reporte se cuenta con dos botones, el que genera los clientes por estado y sus ingresos.

Una forma rápida de visualizar cómo se comporta la empresa en los diferentes estados, información que sirve para atraer más clientes en donde existe poca afluencia, mostrado en la tabla 4.6.4.3.

**Acceso:** Contador/Analista, Gerente y Subgerente.

**Generado:** En el momento que se desee.

Delegación	Total de Clientes
ALVARO OBREGON	2
AZCAPOTZALCO	8
BENITO JUAREZ	3
COYOACAN	2
CUAJIMALPA	1
CUAUHTEMOC	3
GUSTAVO A. MADERO	3
IZTACALCO	5
IZTAPALAPA	2
MAGDALENA CONTRERAS	1

Delegación	Ingresos
ALVARO OBREGON	\$36,500.00
AZCAPOTZALCO	\$13,020.00
BENITO JUAREZ	\$11,070.00
COYOACAN	\$6,000.00
CUAJIMALPA	\$3,000.00
CUAUHTEMOC	\$1,650.00
GUSTAVO A. MADERO	\$4,750.00
IZTACALCO	\$34,270.00
IZTAPALAPA	\$1,400.00
MAGDALENA CONTRERAS	\$1,200.00



MIGUEL HIDALGO	1
MILPA ALTA	1
TLAHUAC	1
TLALPAN	1
VENUSTIANO CARRANZA	1
XOCHIMILCO	1

MIGUEL HIDALGO	\$2,100.00
MILPA ALTA	\$1,050.00
TLAHUAC	\$380.00
TLALPAN	\$4,750.00
VENUSTIANO CARRANZA	\$6,000.00
XOCHIMILCO	\$1,400.00

CLIENTES POR DELEGACION

INGRESOS POR DELEGACION

**Tabla 4.6.4.3 Reporte de Datos Estadísticos de Resumen por Estados.**

**4.6.4.4 Reporte de Datos Estadísticos de Cuentas por Meses.**

Este reporte tiene los campos de los doce meses del año y los tipos de cuenta que se tienen registrados en la institución, es una tabla de referencia cruzada, nos indica en que mes se apertura un tipo de cuenta y cuantas fueron realizadas, únicamente se muestra en pantalla y es una mirada a través de todos los meses del año y su comportamiento de acuerdo al rubro de cuenta, como se puede visualizar en la tabla 4.6.4.4.

**Acceso:** Contador/Analista, Gerente y Subgerente.

**Generado:** En el momento que se desee.



TIPO DE CUENTA	ENE	FEB	MAR	ABR	MAY	JUN	JUL	AGO	SEPT	OCT	NOV	DIC
A PLAZOS	1	1	1	1	1		1					
AHORRO FACIL			1	3	1				3			
AHORRO NORMAL				3	2				2		1	1
CUENTA CORRIENTE		1						1		3		1

**Tabla 4.6.4.4 Reporte de Datos Estadísticos de Cuentas por Meses.**

Con esta última tabla podemos concluir el área de generación de reportes, considerando que son los más importantes para la empresa, en caso de que se solicite algún otro o se desea realizar algún estimado, se harían directamente en la base de datos por medio de queries (consultas) y posteriormente, si es de uso muy común, ingresarlo al sistema para que sea visualizado en tiempo real y poder así administrar y visualizar la información almacenada en la Base de Datos.



## CONCLUSIONES



---

## CONCLUSIONES

A través del proceso de desarrollo del sistema y de acuerdo al objetivo que se planteó podemos establecer lo siguiente:

- El Sistema de Administración y Cuentas de Ahorro cumple con el objetivo planteado al inicio que es el de lograr la administración y control de los productos bancarios que la institución ofrece a los clientes; resolviendo los problemas que sufría la empresa con los métodos anteriores.
- La importancia de crear y desarrollar un sistema es principalmente conocer las necesidades o problemática que presenta el cliente para poder brindarle la solución óptima a dichas necesidades.
- Con el desarrollo de la aplicación se creó una herramienta de software útil eficiente, escalable, competitiva y amigable para el usuario final.
- Gracias a la creación del sistema se evitará almacenar una gran cantidad de papeles y se permitirá reducir el trabajo del personal al llevar el proceso de registro de la información a mano, dando como resultado la obtención de información precisa y actualizada en el momento que se requiere.
- El sistema logra agilizar el análisis de información permitiendo simplicidad y rapidez en las consultas.
- Se cubren las necesidades de seguridad y respaldo de la información importante dentro de la base de datos.
- La implementación del sistema causará buen impacto en los usuarios ya que uno de los objetivos del sistema es el de registrar, almacenar y organizar de una manera rápida y eficiente toda la información de las cuentas de ahorro.
- La implementación de la metodología adecuada se utilizó con el fin de satisfacer las necesidades del cliente y nos guió durante todas las fases de su elaboración para concluir con éxito nuestro proyecto.



- El sistema implementado cumple con las expectativas del cliente, debido a que mejoró sus tiempos de respuesta, además de evitar errores y ganar mayor prestigio entre sus clientes.
- El Sistema aportó grandes mejoras a la Sociedad de Ahorro gracias a los reportes que permiten ver los rendimientos en las cuentas y la toma de decisiones se realiza en forma oportuna y con bases verídicas.
- Utilizar los recursos con los que ya contaba el cliente nos permitió reducir costos de construcción del software además de demostrar que se pueden desarrollar sistemas eficientes y de calidad aprovechando herramientas como Microsoft Access.
- La utilización de herramientas visuales reduce el tiempo de desarrollo agilizando el proceso de entrega de la solución informática, además de que crea una interfaz amigable para el usuario que hace más fácil el manejo del sistema generando un menor tiempo de capacitación y por lo tanto, reducción en los gastos de la compañía.
- Gracias al diseño del sistema se logró beneficiar a la Sociedad de Ahorro al minimizar las pérdidas que reportaba la compañía generando beneficios en la productividad y eficiencia laboral del personal, así como un mejor servicio a sus clientes. Lo anterior, repercute en aumento de la cartera de clientes y mayores ingresos.
- Respecto a los resultados esperados, se cubrieron satisfactoriamente las expectativas de desarrollo y desempeño del sistema, ya que se logró consolidar un sistema completo, práctico y funcional, listo para ser implementado.
- Para el desarrollo del Sistema de Administración y Control de Cuentas de ahorro se llevo a cabo el proceso de análisis, diseño, prueba e implantación de un sistema, lo cual comprueba que los conocimientos adquiridos en las materias que se imparten en la carrera de Ingeniería en Computación proporcionan las



herramientas necesarias para realizar aplicaciones profesionales de una manera sencilla y con un rendimiento aceptable.

- La experiencia obtenida a lo largo de este proyecto de tesis ha sido fundamental para obtener nuevos conocimientos sobre otras áreas como contabilidad, finanzas, operaciones y reafirmar los ya adquiridos.
- Podemos concluir que aunque se realice una tarea varias veces con éxito, el sistema siempre será perfectible y que de todas las pruebas recomendadas para ejecutar sobre un sistema, siempre la más contundente será la que realice el usuario final.

#### Conclusiones Personales

- El proceso de Titulación es un trabajo que debe llevar una metodología que permite crecer en diferentes ámbitos personales, en mi caso, me ayudo a seguir un plan de trabajo y llevarlo rigurosamente, ya que si no se sigue al pie de la letra, se producen cambios en toda la metodología, retrasando otras actividades que se tenían planeadas, de la misma forma, se desarrollo en mí, un gusto particular para la investigación, a ser autodidacta y a la responsabilidad de realizar las cosas de la mejor manera, esto para obtener resultados de calidad.
- Reforcé conocimientos adquiridos en la escuela y en los lugares donde laboré, dándome cuenta que todas las materias sirven para poder realizar trabajos de esta magnitud, permitiendo explotar mi creatividad y desarrollarme en otras áreas que jamás había experimentado, como lo son la planeación basadas en métodos científicos, la aplicación de técnicas de estudio, por mencionar algunas.
- A trabajar basado en objetivos y en forma conjunta, identificando que la comunicación y el saber escuchar, son bases importantes para la realización de cualquier actividad.
- Una satisfacción muy grande al concluir un proyecto aplicable y poder cubrir las necesidades de un cliente al resolver la problemática de su empresa.





## **BIBLIOGRAFIA.**



## Capítulo 1

- Baca Urbina Gabriel, “Evaluación de Proyectos” Mc Graw-Hill. Impreso en México. 4<sup>a</sup>. Edición, 2001. Página: 180.
- Olguín Romo Heriberto, “Dirección, Organización y Administración de Centros de Tecnología de Información” México, UNAM, Facultad de Ingeniería. 1<sup>a</sup>. Edición, Abril de 2005. Páginas: 44 y 64.
- Samuelson, Paul A. y Nordhaus , William D., “Economía”, Ed. McGraw-Hill, México 1992. Págs. 144,158-166.
- <http://www.contabilidadyfinanzas.com/conciliacion-bancaria.html> (Costos de operación y Propuesta del nuevo sistema).
- [http://www.wikilearning.com/monografia/organizacion\\_y\\_administracion\\_de\\_tesoreria-evaluacion\\_de\\_los\\_costos\\_bancarios/13149-5](http://www.wikilearning.com/monografia/organizacion_y_administracion_de_tesoreria-evaluacion_de_los_costos_bancarios/13149-5) (Costos de operación y Propuesta del nuevo sistema).
- <http://www.monografias.com/trabajos14/costosbanc/costosbanc.shtml> (Costos de operación y Propuesta del nuevo sistema).
- [http://www.puntodeequilibrio.com.pe/punto\\_equilibrio/01i.php?bolnum\\_key=17&id=15406&pantalla=noticia&serv\\_key=2100](http://www.puntodeequilibrio.com.pe/punto_equilibrio/01i.php?bolnum_key=17&id=15406&pantalla=noticia&serv_key=2100) (Costos de operación y Propuesta del nuevo sistema).



- <http://www.virtual.unal.edu.co/cursos/sedes/manizales/4100010/Lecciones/Cap4/DFDPasoAPaso.htm> (Costos de operación y Propuesta del nuevo sistema).

## Capítulo 2

- E. Rivero Cornelio, “Bases de Datos Relacionales”, Paraninfo, Páginas 24-32, 45-56.
- Mendelzon- Ale, “Introducción a la base de datos relacionales” Pearson Educación, Páginas 9-50.
- James Martin, “Organización de las Bases de Datos”, Prntice Hall, Páginas 19 y 145.
- Shakuntala Atre, “Técnica de Base de Datos”, Trillas. Páginas 24 y 73.
- C.J.Date “Introducción a los Sistemas de Bases de Datos” Sitsa, Páginas 224-297.
- Ceballos, Fco. Javier,” Curso de programación de Visual Basic 6. (2000)”. Alfaomega.
- Ceballos, Fco. Javier,” Enciclopedia de Microsoft Visual Basic 6. (2000)”.Alfaomega.
- [http://es.wikipedia.org/wiki/Visual\\_Basic](http://es.wikipedia.org/wiki/Visual_Basic) (características, ventajas y desventajas de Visual Basic).



- <http://www.monografias.com/trabajos2/guiavb/guiavb.shtml> (características, ventajas y desventajas de Visual Basic).
- <http://www.monografias.com/trabajos10/visual/visual.shtml> (características, ventajas y desventajas de Visual Basic).
- [www.pericia.cl/Doc/FORMNORMAL.doc](http://www.pericia.cl/Doc/FORMNORMAL.doc) (*formas normales*).
- [http://es.wikipedia.org/wiki/Base\\_de\\_datos\\_relacional](http://es.wikipedia.org/wiki/Base_de_datos_relacional) (bases de datos relacionales).
- [http://es.wikipedia.org/wiki/Base\\_de\\_datos\\_relacional](http://es.wikipedia.org/wiki/Base_de_datos_relacional) (características, ventajas y desventajas de las bases de datos relacionales).
- <http://www.angelfire.com/crazy3/eirita/Investigacion.1.htm> (características, ventajas y desventajas de las bases de datos relacionales).
- <http://www.brighthub.com/computing/windows-platform/articles/41003.aspx> (características, ventajas y desventajas de Access).
- <http://bicosyes.com/benchmark-mysql-vs-postgresql-vs-ruby/> (características, ventajas y desventajas de Access).
- <http://eses.seocounter.net/%C2%BFcu%C3%A1ndo-debe-usted-utilizar-servidor-microsoft-sql-sobre-microsoft-access-15261a.html> (características, ventajas y desventajas de Access).
- <http://eleconomista.com.mx/8-bytes/2011/04/25/que-office-microsoft-sigue-siendo-tan-popular> (características, ventajas y desventajas de Access).



- <http://delphi.about.com/library/bluc/text/uc051001a.html> (características, ventajas y desventajas de Access).
- <http://office.microsoft.com/en-us/access-help/access-specifications-HP005186808.aspx#top> (características, ventajas y desventajas de Access).

### Capítulo 3

- Farley E. Richard, “Ingeniería de Software”, Mc Graw-Hill, Primera Ed. Español, 1992 Pags 40 a 47.
- Pérez, César “Oracle 9i Servidor de Aplicaciones, Red y Programación”, Ed. Alfaomega-Rama.
- Pressman S. Roger, “Ingeniería del Software, Un enfoque Práctico”, Mc Graw-Hill Cuarta Edición, 1998, Pags 22 a 24.
- [http://es.wikipedia.org/wiki/Módulo\\_\(informática\)](http://es.wikipedia.org/wiki/Módulo_(informática)) (Identificar los posibles módulos de la aplicación).
- <http://sigip.blogspot.com/2011/04/justificacion.html> (Justificación de la metodología y del software a utilizar).
- <http://www.mitecnologico.com/Main/ModeloCicloDeVidaClasicoOEnCascada> (Justificación de la metodología y del software a utilizar).



- <http://www.monografias.com/trabajos5/inso/inso.shtml> (Justificación de la metodología y del software a utilizar).

## Capítulo 4

- Rakitin, Steven R. , “SOFTWARE VERIFICATION AND VALIDATION, a Practitioner’ s Guide”, Artech House.
- Schneider, Geri and Winters, Jason P. “Applying Use Cases: A Practical Guide”, Addison-Wesley, 1998.
- Stokes, D. A. “Requirements analysis. Computer Weekly Software Engineers” Reference Book, 1991.
- <http://www.mailxmail.com/curso-sistemas-informativos-diseno-creacion-2/diagrama-contexto-determinacion-tareas-sistema> (Diagramación).
- <http://www.librosdeluz.net/comparacion-entre-sistemas-de-gestion-de-bases-de-datos-sgbd-bajo-licenciamiento-libre-y-comercial-libro-gratis/> (Diagramación).
- <http://www.monografias.com/trabajos59/administracion-diseno-db/administracion-diseno-db2.shtml> (Diagramación).
- <http://www.desarrolloweb.com/faq/452.php> (Diagramación).
- <http://mis-algoritmos.com/aprenda-a-crear-diagramas-de-flujo> (Diagramación).