

12
21.



UNIVERSIDAD NACIONAL AUTONOMA
DE MEXICO

FACULTAD DE INGENIERIA

SISTEMA DE INFORMACION PARA EL
MANEJO DE OPERACIONES BANCARIAS

T E S I S

QUE PARA OBTENER EL TITULO DE
INGENIERO EN COMPUTACION
P R E S E N T A N :

JAVIER BARCENAS ROSAS
FRANCISCO JAVIER GONZALEZ MARTINEZ

DIRECTOR: ING. ADOLFO MILLAN NAJERA

MEXICO, D. F.

1997

TESIS CON
FALLA DE ORIGEN





Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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

Agradecimientos

A la Universidad Nacional Autónoma de México, a la Facultad de Ingeniería y a sus profesores, por habernos dado los elementos para enfrentar a la vida profesional.

Al Ing. Adolfo Millán por su gran compromiso con la Universidad y sus alumnos, así como su cooperación para el desarrollo de este trabajo de tesis.

Javier Bárcenas y Javier González

A mi papá, que con su ejemplo y dedicación ha sido la piedra que formó los pilares de mi carácter.

A mi mamá, que con su cariño y entrega me ha dado fuerza para seguir adelante.

A Gabriela y Efraín mis queridos hermanos.

A María, quien me ha acompañado en este sinuoso camino que es la vida.

A Jairo, Arturo, Guillermo y Javier, hermanos que me dio a escoger el destino.

A Kerigma, que me ha ayudado a conseguir mis objetivos.

Javier Bárcenas

**A mis padres por haberme dado todo
para convertirme en lo que soy.**

**A la familia González Batta, a mis
abuelos y la familia Salgado Martínez
por su compañía y grandes momentos
a lo largo de toda la vida.**

**A Javier por su gran amistad y
paciencia para aguantarme todo este
tiempo.**

**A Guillermo por ser todo un amigo,
por su ayuda y por seguir con
nosotros.**

**A Paco, Pao, Vero y Hebert por ser
grandes compañeros y amigos.**

Javier González

Prólogo

El siguiente trabajo de tesis presenta cada uno de los pasos efectuados para la implementación de un Sistema de Información para el Manejo de Operaciones Bancarias, es decir, un medio alternativo a través del cual los clientes de una institución bancaria podrán realizar operaciones de una manera diferente a como se han estado desarrollando con la ventaja de ser más económico para el banco y más rápido y fácil de uso para los clientes.

Este documento comprende una introducción en donde se presenta la situación actual del sector bancario y la necesidad de una solución que permita al banco atender a más clientes a un costo menor y así mantenerse a un nivel competitivo. Una vez en la parte de análisis se presentan algunos de los puntos clave que se deben atacar en la atención de clientes en sucursal. Con todo esto se plantea como solución una terminal de información con gran facilidad de uso y bajo costo.

Para la implementación de este sistema se eligió la metodología de lo general a lo particular y se presenta la teoría para cada uno de los pasos a seguir. Con esto, se genera el diagrama de flujo de datos que es la parte final del proceso de análisis.

El apartado que se refiere al diseño presenta los elementos de la teoría del diseño estructurado para la generación de la carta estructurada del sistema de información y a continuación se muestran ejemplos de las especificaciones técnicas para la programación de los módulos. Por último se presentan los estándares de diseño, las entradas, salidas, controles y procedimientos del sistema.

El desarrollo explica las características tomadas en cuenta para la elección del lenguaje de programación y hace una revisión de cada uno de los elementos de hardware que componen la terminal y su relación con el software, así como las partes clave en el desarrollo del sistema.

En la etapa de prueba se presentan los diferentes métodos para evaluar un sistema una vez terminado y se señalan cada uno de los eventos presentados cuando se le aplicó la prueba al sistema.

En la implantación se señalan los diferentes métodos de conversión y se indica el método seleccionado con los apuntes sobre las experiencias adquiridas.

Por último en las conclusiones se detallan las observaciones hechas cuando el sistema fue puesto en producción al público y las expectativas que se presentan para este desarrollo en el futuro.

Prólogo	i
1. Introducción	1
1.1 INTRODUCCIÓN	1
1.2 OBJETIVO	2
1.3 ANTECEDENTES	3
1.4 ALCANCE	3
1.5 SOLUCIÓN	4
2. Análisis	6
2.1 PLANTEAMIENTO DEL PROBLEMA	6
2.2 SOLUCIÓN	10
2.3 METODOLOGÍA	16
2.3.1 METODOLOGÍA DE LO GENERAL A LO PARTICULAR	16
2.3.2 ESTRATEGIA DE FLUJO DE DATOS	16
2.3.3 ESTRATEGIA DEL ANÁLISIS DE DECISIONES	17
2.3.4 CONCEPTO DE FLUJO DE DATOS	18
2.3.5 NOTACIÓN	19
2.3.6 ACTIVIDADES PARALELAS	21
2.3.7 VENTAJAS DEL MÉTODO	22
2.3.8 VISIÓN LÓGICA DE LOS SISTEMAS	23
2.3.9 TRABAJO DE LO GENERAL A LO PARTICULAR	24
2.3.10 EXTENSIÓN DE PROCESOS PARA MAYOR DETALLE	25
2.3.11 MANTENER CONSISTENCIA ENTRE PROCESOS	29
2.3.12 AÑADIR CONTROL SOLAMENTE EN LOS DIAGRAMAS DE MENOR NIVEL	29
2.3.13 ASIGNAR ETIQUETAS SIGNIFICATIVAS	31
2.3.14 NOMENCLATURA DEL FLUJO DE DATOS	31
2.3.15 NOMENCLATURA DE LOS PROCESOS	32
2.4 FLUJO DE DATOS	33
2.4.1 INTRODUCCIÓN	33
2.4.2 DIAGRAMA DE FLUJO DE DATOS	38
3. Diseño	47
3.1 ESPECIFICACIONES DE LOS ELEMENTOS LÓGICOS DEL DISEÑO	47
3.2 DISEÑO ESTRUCTURADO	49
3.3 PROPORCIONAR ESPECIFICACIONES DETALLADAS PARA EL DESARROLLO DE SOFTWARE	56
3.4 AJUSTARSE A LOS ESTÁNDARES DE DISEÑO	61
3.5 DISEÑO DE LA SALIDA	67
3.6 DISEÑO DE LA ENTRADA	69
3.7 DISEÑO DE CONTROLES	72
4. Desarrollo	74

4.1 INTRODUCCIÓN	74
4.2 PROGRAMACIÓN Y DESARROLLO	76
5. <i>Etapas de Pruebas</i>	95
5.1 ASEGURAMIENTO DE LA CALIDAD	95
5.2 NIVELES DE SEGURIDAD	95
5.2.1 PRUEBA	95
5.2.2 VERIFICACIÓN Y VALIDACIÓN	96
6. <i>Implantación</i>	103
6.1 MÉTODOS DE CONVERSIÓN	103
6.2 SISTEMAS PARALELOS	103
6.3 CAMBIO DIRECTO	104
6.4 ENFOQUE PILOTO	105
6.5 MÉTODO POR ETAPAS	106
6.6 PLAN DE CONVERSIÓN	108
6.7 ANÁLISIS DE IMPACTO	109
7. <i>Conclusiones</i>	112
<i>Bibliografía</i>	117

Capítulo 1

Introducción

1.1 INTRODUCCIÓN

Antiguamente los bancos basaban su negocio exclusivamente en actividades crediticias, es decir, el banco prestaba el dinero a un interés mayor al que pagaba a sus clientes por el manejo de su capital. Debido a que en los últimos años las necesidades del cliente han cambiado hacia una demanda mayor de servicios (por ejemplo pagos de nómina, pago de servicios públicos, operaciones bancarias, etc.) y el sentimiento de lealtad hacia la institución se ha perdido, los bancos tienen que cambiar la visión del negocio y ofrecer al cliente una gama de soluciones para cubrir todos sus requerimientos.

El cliente tiene necesidad de hacer varias operaciones en el menor tiempo posible. Éstas actualmente son realizadas en un mostrador, lo cual representa filas en la sucursal, personal atendiendo y horarios reducidos. El cajero automático se presenta como una alternativa a estas operaciones, sin embargo dicho instrumento es caro por la seguridad que requiere, voluminoso y pesado por la caja fuerte donde guarda el dinero y para el público en general es difícil aceptar el uso de este equipo ya que su cultura no le permite entender el funcionamiento de este servicio.

Por lo anterior se ha introducido al mercado una terminal de información, que es un dispositivo que ofrece las características operativas del cajero automático, además de las ventajas que representa un fácil uso, atractivo al cliente, menor tamaño y costo, entre otras, pero sin manejo de efectivo ya que muchas de las transacciones bancarias no requieren de éste.

Se propone que esta terminal de información maneje un sistema que permita que el cliente sea atraído por el equipo, ya que utiliza video y sonido estéreo. Una vez captada su atención se le invita a que haga uso al sistema a través de una pantalla sensible al tacto.

Las operaciones que el sistema puede realizar son: transferencia de fondos, pago de servicios, consulta de saldos, consulta de datos del cliente, consulta de información financiera, impresión de estados de cuentas, etc.

Las terminales también pueden servir como herramienta de mercadotecnia, dando información al cliente de servicios que ofrece el banco, evaluación de un cliente para ver si es sujeto de crédito, localización de sucursales, promociones, entre otras.

1.2 OBJETIVO

Elaborar un sistema de atención personal que permita mostrar parte de la gama de productos y servicios que una institución bancaria puede ofrecer (inversión, ahorro, crédito,

pago de servicios), así como un servicio de operaciones bancarias (consultas, transferencias, pagos de servicios, información financiera, cambio de clave secreta, etc.).

1.3 ANTECEDENTES

En algunas instituciones bancarias se tenía un servicio de información con el cual el cliente sabía sobre sus tarjetas de crédito, así como información financiera, pero el equipo era muy poco llamativo y con pocas variantes, además de que no proporcionaba información de créditos hipotecarios o automotrices entre otros.

1.4 ALCANCE

Los módulos que se desarrollarán serán los siguientes:

Operaciones Bancarias:

- Consultas
- Transferencias
- Pagos de Servicios
- Operaciones Financieras
- Cambio de Clave Secreta

Productos

- Cuenta de cheques

- Tarjeta de Inversión
- Créditos
- UDIS
- Cuenta Única

Servicios

- Electrobanca
- Telebanca
- Multicaja
- Caja Divisas

1.5 SOLUCIÓN

Se empleará un equipo interconectado con la computadora central del banco (Servidor) mediante un equipo de comunicaciones.

El sistema en el equipo será programado en Visual Basic, y en el Servidor habrá un manejador de transacciones CICS dedicado a habilitar instrucciones que disparan programas en COBOL que realizan las operaciones de consulta o transferencias de información.

Mediante este sistema la institución bancaria podrá ofrecer al cliente un servicio de información que le permita rápidamente pagar servicios como luz, agua, teléfono, etc., transfiriendo desde una cuenta de cheques y realizar transacciones bancarias como transferencia de fondos, obtención de información financiera, consulta de saldos, etc. Además, esto incrementará la afluencia de personas al banco ya que transacciones que antes les llevaba mucho tiempo, les tomarán ahora tan sólo unos minutos. Ellos buscarán la información que les permita saber si son sujetos de crédito o no y los requisitos para los diferentes productos de crédito, sin la necesidad de la ayuda de un ejecutivo de sucursal. Toda la información será obtenida con mayor rapidez, confianza y eficiencia.

Capítulo 2

Análisis

2.1 PLANTEAMIENTO DEL PROBLEMA

Debido a que en tiempos recientes la competencia entre los bancos se ha incrementado, cada uno de ellos tiene que presentar al público diversas formas de captar su atención, esto nos lleva a crear productos basados en nuevas tecnologías. Ya que por su rapidez, exactitud, eficiencia, etc., se va creando una gama inmensa de soluciones a las necesidades que se presentan.

Se tratará de no tener tanto tiempo al cliente en una sucursal, para realizar operaciones o pedir informes por la diversidad de productos que ofrece el banco, a un precio menor a lo que actualmente cuesta con otras opciones.

Hay necesidad de agilizar al cliente el trámite de consulta, pago de tarjeta, pago de servicios como agua, luz, teléfono, etc., transferencia entre cuentas, además de tener un medio que pueda informar sobre los diversos productos que ofrece el banco, con todo esto se consigue captar al cliente pues este podrá hacer sus operaciones más rápido sin necesidad de formarse en filas que quitan mucho tiempo, y sin tener tantos funcionarios especializados que tardan en atenderlo.

Según datos proporcionados por la División de Banca de Servicio del banco, al realizar estudios sobre el comportamiento de la afluencia de clientes a una sucursal se obtuvo que en promedio, diariamente se presentan alrededor de 25 personas a pedir información sobre servicios, ya sea requisitos o características, para proporcionar este servicio, a un funcionario le toma de 10 a 15 minutos, así que un cliente podría esperar hasta 15 minutos para ser atendido.

Haciendo una cronología de un día desde que abre la sucursal hasta que cierra, en promedio el flujo de personas que llega a la sucursal a pedir informes es el siguiente:

a)	9 AM-10 AM	2 personas
b)	10 AM-11 AM	5 personas
c)	11 AM-12 PM	6 personas
d)	12 PM-13 PM	7 personas
e)	13 PM-14 PM	3 personas
f)	14 PM-15 PM	2 personas

En una sucursal se tienen por lo menos 3 funcionarios que dan informes y además realizan otras operaciones, con base en las cifras anteriores se demuestra que esta cantidad de funcionarios no se da abasto para atender a todo el público cuando éste se presenta.

La información que recibe un cliente en cuanto a requisitos o características acerca de un producto siempre es la misma y sólo varía muy poco dependiendo de las necesidades y recursos de éste.

Los productos que ofrece el banco son: Cuenta Única, Tarjeta de Inversión, Cuenta de Cheques, Tarjeta de Crédito, Cuenta de Ahorro, Créditos y UDIS, y de los que se pide más información son Tarjeta de Crédito, Cuentas de Cheques y Ahorro.

El 20% de las personas que acuden a una sucursal es para pagar servicios públicos, esperando a veces hasta 30 minutos para poder realizar su operación en días de quincena y horas pico. Aproximadamente el 50% de estas personas tienen cuenta de algún tipo en el banco a la cual se le puede cargar el importe de la operación.

Una alternativa que se ofrece son los cajeros automáticos, pero éstos son muy costosos, no son atractivos y son poco amigables. Debido a la manera de pensar del público en general es necesario un equipo llamativo que lo invite a operarlo de una manera fácil, para que pierda el miedo a descomponerlo o que su transacción no se efectúe correctamente.

En horas pico un cajero automático puede atender aproximadamente a 40 personas, pero aún así es alto el porcentaje de gente que prefiere ir a la ventanilla. Esto va en perjuicio del banco ya que una operación en ventanilla cuesta aproximadamente 2.00 dólares y en cajero 1.50 dólares.

El precio de un cajero se incrementa debido a que se necesita tener una gran seguridad para controlar todos los dispositivos que manejan el dinero (cajones de dinero, contadores de dinero, puerta para entregar el dinero, bandas transportadoras y fotosensores).

Las razones que se plantearon anteriormente nos permiten ver las áreas de oportunidad que se pueden mejorar en el banco INVERLAT. Los clientes están esperando demasiado tiempo en la sucursal, lo cual hace que el cliente busque otra opción y deje de ir a las sucursales del banco INVERLAT. Los cajeros automáticos son poco amigables, costosos y además es muy caro su mantenimiento. Los ejecutivos en una sucursal dedican gran parte de su tiempo dando informes repetitivos, que fácilmente podría darlos una computadora, además de hacerle cálculos personalizados al cliente para ver si puede aspirar a un crédito de casa, auto etc.; y no hay forma de que el cliente que tenga cuenta en el banco pueda realizar transacciones de compra y venta de títulos bancarios sin tener que pedir ayuda de un ejecutivo dedicado a ese tipo de operaciones.

2.2 SOLUCIÓN

Una solución son las terminales de atención electrónica, como los cajeros automáticos y la terminal de información para manejo de operaciones bancarias.

Otra institución bancaria, Banamex, tiene en sus sucursales una terminal al servicio del público, en la cual se pueden consultar saldos, información y transferencia de fondos, además de poder hacer otras operaciones. La desventaja es que la terminal no es muy atractiva al cliente, el cual no la utiliza tan frecuentemente como se quisiera, además que el monitor sensible al tacto no funciona muy bien, por lo que el cliente termina por desesperarse e irse.

Con un equipo que incorpore elementos multimedia podríamos superar muchos de los problemas que se presentan hoy en día con los viejos sistemas. La terminal de información para manejo de operaciones bancarias puede presentar videos cortos con música estéreo, comerciales de televisión o animaciones que llaman la atención del público, además de reducir los costos de operación de diversas transacciones.

El costo de la transacción en esta terminal sería de 1.00 dólar, éste fue determinado por el proveedor basándose en instalaciones similares existentes en Chile. Además la relación de precio entre esta terminal y los cajeros es muy alta, ya que un cajero automático tiene en valor de 14,000 a 18,000 dólares por que necesita una caja de seguridad donde se

guarda el dinero, y dispositivos para el control de éste, que incrementan mucho el costo; el precio de la terminal es solamente de 8,000 dólares reduciendo un 50% los costos actuales.

Algunas ventajas más que puede ofrecer el sistema son: facilidad de uso, atractivo al usuario, reducción en el tiempo de filas, menos pérdida de tiempo proporcionando informes.

De manera general el funcionamiento se describirá a continuación: el sistema despliega en una terminal de información una serie de menús que permiten al usuario recibir información y realizar operaciones bancarias comunicándose con una tarjeta Ethernet a la red de la sucursal. Cuando no se está utilizando, periódicamente presenta videos de gran calidad y con música estéreo para atraer clientes. Se puede leer tarjetas de banda magnética, imprimir recibos y leer códigos de barras para el pago de servicios como teléfono y luz con transferencias de cuentas de cheques.

Esta terminal combina elementos de tecnología nueva como es la PC 486, Visual Basic y elementos totalmente innovadores como la tarjeta para descomprimir el video y la tecnología de la pantalla sensible al tacto

En este momento cada sucursal cuenta con una red conectada al servidor central, a la cual se puede conectar la terminal de información, para que desde ésta se puedan disparar transacciones de consulta de saldo, transferencia de fondos, pago de servicios públicos, etc.

El equipo en la sucursal podrá realizar operaciones sobre los sistemas de clientes, cheques y concentración de fondos que se encuentran en el equipo central.

Como se mencionó anteriormente la terminal nos servirá también para dar información de diversos artículos bancarios ya que la mayor parte de ésta siempre es la misma o varía en porcentajes que se calculan con fórmulas que se pueden programar y no es necesaria la intervención de una persona para realizarlas, además de que le ofrece más confianza al proporcionar datos confidenciales que no le sería tan fácil revelar a un empleado.

El sistema se podría expandir a otros servicios, como el pago de televisión por cable, colegiaturas, autofinanciamientos, etc., dependiendo de la aceptación que vaya teniendo. También sería posible incorporar transacciones de otros productos del banco que en esta etapa no están contemplados y otra opción es, presentar la información en inglés u otros idiomas para sucursales que así lo requieran, como sería el caso de aeropuertos y centros turísticos. Una innovación que se pretende para la siguiente etapa es el poder abrir una cuenta por medio del sistema.

El tiempo de realización del proyecto se considera de cuatro meses contando con dos personas para su desarrollo.

ETAPA	TIEMPO
Análisis	2 semanas
Diseño	4 semanas
Desarrollo	6 semanas
Pruebas	3 semanas
Implementación	1 semana
TOTAL	16 semanas

La etapa de liberación a todas las sucursales contemplará las siguientes operaciones: pago de servicios, transferencias, consulta de información, consulta de saldos, compra venta de títulos. Que se llevaría a cabo durante un mes en dos sucursales piloto. Dependiendo de la aceptación se implantaría en todas las sucursales evaluando la respuesta del público a cada uno de los productos que se ofrecieron.

El número esperado de personas que accedan este servicio será proporcional al de clientes que acuden a la sucursal y aquellos con cuentas en el banco. El sistema captará pronto la atención del público debido a todas las ventajas mencionadas anteriormente.

El mantenimiento para este equipo no tiene ninguna complicación ya que es el que se le da comúnmente a una PC, debido a que el único elemento diferente es la pantalla

sensible al tacto que tiene una vida útil de 10 años. Lo que sería necesario es limpieza del equipo, es decir monitor, CPU, lectora de tarjetas, además, se tiene que revisar la cantidad de papel de la impresora, para poner más en el momento que se acabe, la terminal cuenta con un sensor que indica cuando el papel está bajo y otro cuando ya se terminó.

Los videos que se presentan en la terminal pueden ser los mismos que se transmiten por la televisión o animaciones por computadora utilizando música estereofónica que sonará a través de dos bocinas integradas a la terminal.

Como otras ventajas para el banco es posible personalizar los colores del equipo, ya que es importante que la gente reconozca a INVERLAT como el banco de color azul porque esto permitirá diferenciar este servicio del de otros bancos, por otra parte es muy transportable, de fácil manejo y no ocupa mucho espacio en la sucursal, ya que se empotra en la pared (Ver Figura 1). El tamaño del monitor es de 9 pulgadas para permitir privacidad al cliente al realizar sus operaciones.



Figura 1

2.3 METODOLOGÍA

2.3.1 METODOLOGÍA DE LO GENERAL A LO PARTICULAR

Los sistemas de información por computadora normalmente están integrados por muchos componentes. En la mayor parte de los casos, es difícil para los analistas entender todos los componentes a un mismo tiempo; por lo tanto, los investigadores tienen que comenzar con preguntas de tipo general en relación con el propósito del sistema, sus entradas y sus salidas y los procesos incluidos. Gradualmente, al estudiar funciones principales del sistema en forma independiente de otras, los analistas conocen más detalles. En los grandes proyectos de sistemas, varios analistas llevan a cabo una investigación en forma seccionada que la distribuyen entre ellos mismos, de manera que cada uno pueda trabajar en forma independiente, pero que se comuniquen los datos importantes de información a los otros.

Existen dos estrategias ampliamente utilizadas para determinar los requerimientos de información, el flujo de datos y las estrategias de análisis de decisión.

2.3.2 ESTRATEGIA DE FLUJO DE DATOS

Los analistas desean conocer las respuestas para cuatro preguntas específicas: ¿Qué procesos integran el sistema? ¿Qué datos se utilizan en cada proceso? ¿Qué datos se

almacenan? y ¿Qué datos entran y salen del sistema? La importancia se da claramente en el *análisis de datos*.

Los datos manejan las actividades del negocio. Pueden iniciar cursos de acción (por ejemplo, nuevos datos de pedidos de ventas) y pueden procesarse para proporcionar información útil al personal que desea saberlos (por ejemplo, que tan bien se han manejado los hechos) al medir la calidad del trabajo y su porcentaje, utilidad, etc.). El análisis reconoce el papel central de los datos del negocio de la empresas. Cuando se sigue su flujo a través de los procesos del negocio, que es el propósito del análisis del flujo de datos, le indica a los analistas una gran cantidad de datos sobre como se están llevando a cabo los objetivos de la compañía. Al manejar las transacciones y completar las tareas, los datos de entrada se procesan, almacenan, consultan, utilizan, modifican y se emiten. El análisis de flujo de datos estudia el uso de éstos en cada actividad; documenta los hallazgos en los diagramas de flujos de datos, que muestran gráficamente la relación entre los procesos datos, y en los diccionarios de datos, que describen formalmente los datos de sistemas y donde se utilizan.

2.3.3 ESTRATEGIA DEL ANÁLISIS DE DECISIONES

La estrategia del análisis de decisiones para la determinación de los requerimientos de sistemas de información completa el análisis de flujo de datos. Esta estrategia realiza el estudio de los objetivos y llevar a cabo el negocio. Las decisiones se presentan tanto en los niveles operatorios como en los de alto nivel gerencial. La estrategia de análisis de decisión

guía estas preguntas: ¿Qué condiciones pueden surgir para que afecten la decisión? ¿Cuáles son las alternativas de acción disponibles para quien toma la decisión? ¿Qué condiciones son más importantes al tomar la decisión sobre las acciones? Esta estrategia revela los datos y las necesidades de información de las personas que tienen que tomar decisiones específicas.

Con frecuencia el análisis de decisiones también se utiliza por parte de la alta gerencia para la toma de decisiones. La alternativa que seleccionan los gerentes responsables de la toma de decisiones, en cuanto a una estrategia de precios entre un conjunto de alternativas, se maneja en forma diferente a la acción que toma un supervisor de departamento para aceptar o rechazar pedidos. En el primer caso, los analistas deben enterarse sobre los aspectos de precios cuando identifican los objetivos gerenciales y las decisiones que tienen que hacer.

La decisión de rechazar pedidos generalmente ocurre con más frecuencia, de manera que las condiciones y acciones normalmente se conocen. Estudiar la decisión puede no requerir el desarrollo y prueba de nuevos modelos, pero determinar cómo realizar la decisión, sí es un aspecto importante.

2.3.4 CONCEPTO DE FLUJO DE DATOS

El análisis de flujo de datos examina el uso de éstos para llevar a cabo procesos específicos de negocios dentro del objetivo de una investigación de sistemas. Los

diagramas de flujo de datos muestran su uso en el sistema en forma gráfica; es decir, todos los componentes esenciales del sistema y cómo se relacionan entre sí. Puede ser difícil entender por completo un proceso de negocios a través de una descripción verbal solamente; por lo tanto, los diagramas de flujo de datos ayudan a ilustrar los componentes esenciales de un proceso y la forma en la que interactúan.

2.3.5 NOTACIÓN

Los diagramas de flujo de datos lógicos pueden completarse al utilizar solamente tres notaciones sencillas.

Flujo de datos. Los datos cambian en una dirección específica, desde su origen hasta su destino, en forma de un documento, carta, llamada telefónica o cualquier otro medio. El flujo de datos es un “paquete” de datos.

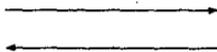


Figura 2.3.1 Líneas de flujo de datos.

Procesos. Procedimientos o dispositivos que utilizan o producen datos.

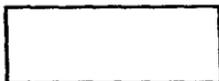


Figura 2.3.2 Rectángulo para representar procesos.

Origen o destino de los datos. El origen externo o destino de los datos, que pueden ser pantallas, reportes o archivos.



Figura 2.3.3 Computadora que representa origen o destino de datos.

Datos almacenados. Aquí los datos se almacenan o se hace referencia a ellos a través de un proceso



Figura 2.3.4 Cilindro que representa datos almacenados.

En el diagrama cada componente de flujo de datos se etiqueta con un nombre descriptivo. Los nombres del proceso se identifican aún con un número que se utilizará para

propósitos de identificación. El número asignado para un proceso específico no representa la secuencia del mismo; es estrictamente para identificación y tendrá otro valor cuando se estudien los componentes que conforman un proceso específico.

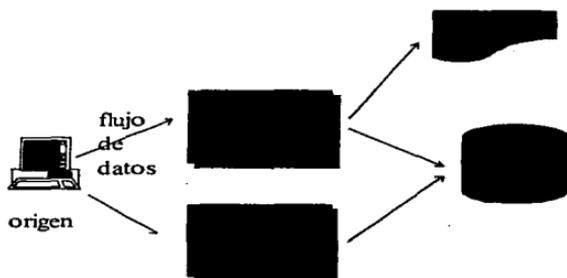


Figura 2.3.5 Ejemplo de diagrama de flujo de datos.

2.3.6 ACTIVIDADES PARALELAS

Nótese en la figura 2.3.5 que pueden estar realizando en forma simultánea varios flujos de datos. Los flujos de datos uno y dos pueden presentarse en forma *paralela*. Esta característica de los diagramas de flujo de datos, para mostrar actividades paralelas, es una ventaja adicional. Otros métodos para diagramar, como los diagramas de flujo, muestran procesos *en serie*, actividades que solamente pueden darse en orden específico una después de otra. Las compañías tienen muchas actividades que aparecen al mismo tiempo con flujos

de datos en forma concurrente. Los diagramas de flujo de datos permiten al analista representar las actividades con mayor exactitud mostrándolas en forma simultánea cuando surgen.

Como su nombre lo sugiere, los diagramas de flujo de datos se concentran en los *datos* que se mueven a través del sistema y no en los dispositivos o el equipo. Por medio del proceso del entendimiento del área de aplicación, los analistas identifican y describen los datos que se mueven a través del sistema, por qué se introducen o retiran y qué proceso se está llevando a cabo. Es muy importante determinar cuando entran o se retiran los datos del área de aplicación. Algunas veces los datos se almacenan para su uso posterior o su consulta en almacenamientos previos. Los diagramas de flujo de datos también muestran lo anterior.

2.3.7 VENTAJAS DEL MÉTODO

Estas notaciones sencillas son fácilmente entendidas por los usuarios y personas de negocios que son parte del proceso que se estudia; por lo tanto, los analistas pueden trabajar con los usuarios y realmente hacen que participen en el estudio del flujo de datos. Los usuarios pueden hacer sugerencias sobre modificaciones de los diagramas para describir en forma más exacta la actividad de los negocios; también pueden examinar los diagramas y señalar los problemas rápidamente de manera que se puedan corregir antes de que se inicie otro trabajo de diseño. Si los problemas no se localizan en forma anticipada en el proceso

de desarrollo, será muy difícil corregirlos posteriormente; incluso evitarlos a tiempo puede prevenir una falla del sistema.

El análisis del flujo de datos permite a los analistas aislar las áreas de interés de la compañía, estudiarlas y examinar los datos que entran al proceso y ver cómo se modifican cuando salen del mismo. Conforme los analistas recopilan hechos y detalles, su creciente conocimiento del proceso los lleva a realizar preguntas sobre partes específicas del proceso, que a su vez los conduce a una investigación adicional.

Una investigación de sistemas elaborada en forma amplia produce conjuntos de muchos diagramas de flujos de datos, algunos de los cuales proporcionan una visión general de los procesos principales, y otros dan un mayor detalle para señalar elementos de datos, almacenamientos de éstos y etapas del proceso para componentes específicos de un sistema mayor. Si los analistas desean revisar posteriormente el sistema en su totalidad, utilizan los diagramas generales. Sin embargo, si están interesados en estudiar un proceso en particular, utilizan los diagramas de flujo de datos de los procesos de menor nivel.

2.3.8 VISIÓN LÓGICA DE LOS SISTEMAS

Los diagramas de flujo de datos muestran las características lógicas de las aplicaciones; es decir, señalan qué ocurre y cuándo, pero no establecen cómo; en otras palabras, los detalles físicos son métodos de almacenaje, dispositivos de entrada o componentes de la computadora, o sea, no son aspectos principales para los analistas

durante la parte inicial del análisis; sólo después de entender los componentes lógicos, deben estudiarse los componentes físicos.

2.3.9 TRABAJO DE LO GENERAL A LO PARTICULAR

Los primeros pasos en la determinación de requerimientos, como se ha indicado, se llevan a cabo para conocer las características generales de los procesos del negocio que se investigan. La parte de más alto nivel de los detalles, por llamarlo así, también se estudia. Conforme los analistas entienden mejor esos detalles, profundizan y recopilan más información específica y detallada. Se realizan preguntas específicas a un nivel de detalle cada vez mayor; este proceso se conoce como análisis descendente.

El ejemplo a continuación mostrará el desarrollo de los diagramas de flujo de datos y cada paso estará descrito en cada una de las siguientes secciones.

Supóngase que un analista que inicia un trabajo sobre un sistema de cuentas por pagar, está interesado en conocer los elementos con los que trabaja el sistema (entradas) y lo que produce (salidas). Seguramente se preguntará sobre la referencia de los datos en el proceso de las cuentas por pagar, que son reclamadas por vendedores y proveedores a quienes su compañía les da dinero por los servicios o mercancías que han proporcionado.

El diagrama de flujo de datos de la figura 2.3.6 describe el proceso de cuentas por pagar a nivel muy general. Este diagrama muestra que los proveedores introducen facturas

y reciben cheques de la compañía. Este proceso de cuentas por pagar requiere cuentas por pagar y datos de los proveedores. Nótese que cada flecha, que representa el flujo de los datos, está etiquetada para mostrar que datos se están utilizando. Los datos del saldo se consultan en los datos de cuentas por pagar que están almacenados para cada proveedor a quien se le ha pagado, y la dirección de éste se consulta en el almacenamiento de datos de proveedores.

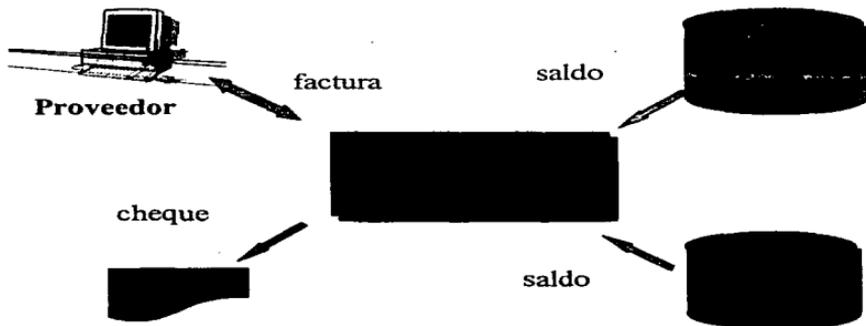


Figura 2.3.6 Nivel general del diagrama de flujo de datos del sistema de Cuentas por pagar.

2.3.10 EXTENSIÓN DE PROCESOS PARA MAYOR DETALLE

La información contenida en la figura 2.3.6 es inadecuada para entender los requerimientos de sistemas. El siguiente paso en la investigación es describir el proceso de

cuentas por pagar con mayor detalle. Las figuras 2.3.6, 2.3.7, 2.3.8 ha "extendido" el proceso de cuentas por pagar a tres subprocesos: APROBACIÓN DE FACTURAS; REVISIÓN DEL SALDO VENCIDO Y ESCRITURA DE CHEQUES. Cada uno es parte del proceso de cuentas por pagar identificado en el de más alto nivel de la figura 2.3.6.

Los números 1.0, 2.0, 3.0 también se utilizan para identificar los procesos; éstos pueden tener como referencia su número (como 1.0) o su nombre (como APROBACIÓN DE FACTURAS).

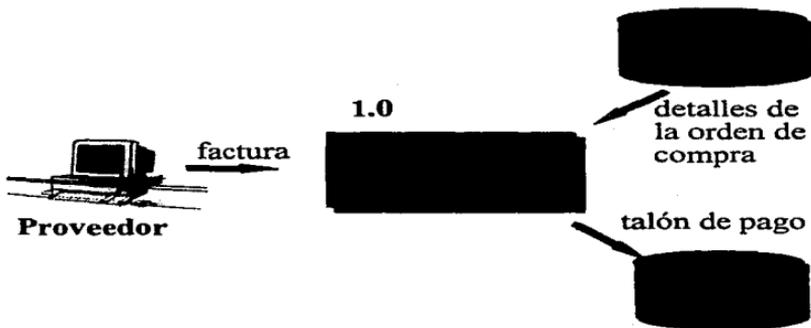


Figura 2.3.7 Aprobación de facturas.

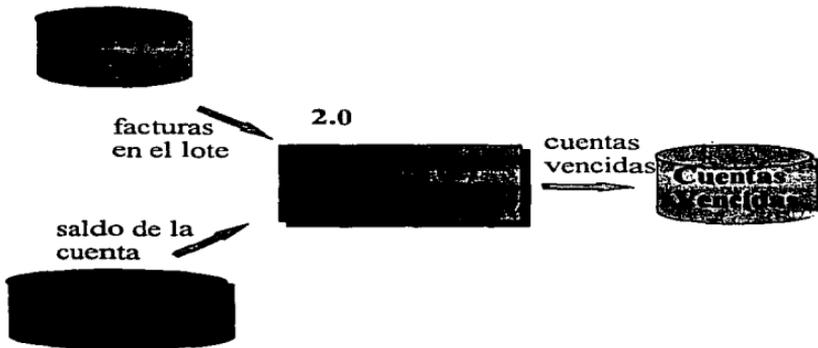


Figura 2.3.8 Revisión del saldo vencido

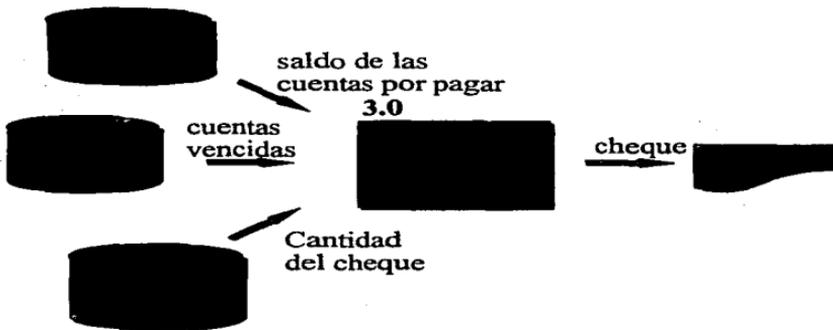


Figura 2.3.9 Escribir los cheques del proveedor

Este paso de descripción de lo general a lo particular (descendente se repite muchas veces en una investigación de sistemas: el entendimiento a un nivel detalle es extendido a un detalle en el siguiente nivel. Las figuras 2.3.7, 2.3.8 y 2.3.9 muestran el proceso original extendido a tres subprocesos, los cuales añaden datos para entender el proceso de cuentas por pagar. En los grandes sistemas, un solo proceso se puede extender muchas veces hasta que se describe una cantidad adecuada de datos para los analistas, y de esta manera puedan entender el proceso en su conjunto.

Los diagramas de flujo de datos no serán útiles si no se dibujan en forma apropiada o son difíciles de manejar. Cada nivel inferior se define utilizando de tres a siete procesos para extender un proceso de mayor nivel. Utilizar más de siete provoca que el diagrama será difícil de manejar. Los diagramas de flujo de datos son más fáciles de leer si la descripción de un proceso de un nivel puede dibujarse en una sola hoja de papel. La guía para utilizar de tres a siete para extender los procesos de mayor nivel deberá permitir que la descripción ocupe una sola hoja de papel. Si se necesitan más datos el siguiente nivel puede extenderse.

El flujo de datos entre los procesos es como un sobre o paquete que contiene uno o más elementos de datos. El sobre contiene todos los datos que fluyen conjuntamente entre dos procesos cada vez; sin embargo, si los diferentes datos cambian a diferentes tiempos, existen dos flujos de datos separados. El flujo de datos TALÓN DE PAGO de la figura 2.3.7 incluye los datos de la factura suministrados por el proveedor y los datos de autorización interna de compra; siempre van juntos y así se incluyen en el sobre de TALÓN

DE PAGO. Sería incorrecto mostrar dos flujos de datos a partir del proceso 1.0 aprobación de la factura.

2.3.11 MANTENER CONSISTENCIA ENTRE PROCESOS

Nótese en la figura 2.3.7 que el primer proceso tiene la misma entrada (factura del vendedor) que la figura 2.3.6. La salida del proceso de escritura de cheques también tiene que concordar con la salida de la figura 2.3.6; por lo tanto, la extensión es consistente; en general, no se introducen en el nivel inferior nuevas entradas o salidas en el proceso que no se identificaron en los diagramas de mayor nivel. Sin embargo, dentro del proceso, se identifican nuevos flujos de datos para "factura", "factura en lote" y "cuentas vencidas" se identificaron al extender el proceso de cuentas por pagar. Este es precisamente el objetivo de la extensión de menos nivel; encontrar más sobre los procesos internos.

2.3.12 AÑADIR CONTROL SOLAMENTE EN LOS DIAGRAMAS DE MENOR NIVEL.

Los diagramas desarrollados hasta este punto no incluyen información de control. No se ha hecho ninguna mención de cómo manejar los errores o excepciones, como la entrada al proceso de las facturas que están incompletas. Aunque esta información es necesaria en el análisis final, no debe importar mientras se está identificando el flujo de datos en general. Los diagramas secundarios (bajo el segundo o tercer nivel) muestran un manejo de errores y de excepciones en el proceso que se extiende.

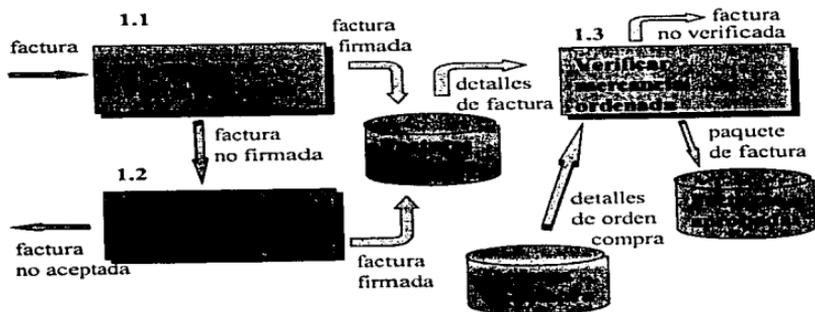


Figura. 2.3.10 Aprobación de la factura I

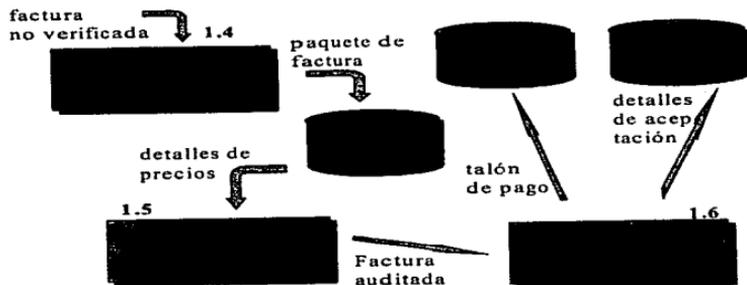


Figura 2.3.11 Aprobación de la factura II

La figura 2.3.10 y 2.3.11 es el diagrama secundario de flujo de datos para el proceso (aprobación de facturas) de la figura 2.3.7. La figura anterior no toma en cuenta tales excepciones como facturas no firmadas o facturas autorizadas sin los pedidos de compra. Aunque estas condiciones son importantes en el proceso de corrección de factura, son detalles específicos del proceso de facturas y por lo tanto se añaden a la extensión de ese proceso.

2.3.13 ASIGNAR ETIQUETAS SIGNIFICATIVAS

Las descripciones asignadas a los flujos de datos y procesos deben señalar al lector que está sucediendo. Todos los flujos de datos se deben nombrar para reflejar su contenido en forma exacta.

2.3.14 NOMENCLATURA DEL FLUJO DE DATOS

Los nombres asignados a los flujos de datos deben reflejar los datos de interés para los analistas, no el documento en el cual se reciben. Algunos sistemas, como aquellos utilizados por las computadoras para realizar el proceso en línea, pueden evitar totalmente el papel. Por ejemplo, una factura contiene muchos diferentes elementos de información. Los analistas están interesados en el contenido de la factura que es importante para un proceso específico. Puede ser el número de facturas, la fecha de expedición o la firma de autorización que va asociada con la misma. No es la hoja de papel (si los analistas diseñan

posteriormente un sistema sin incluir papel, se omitirá la hoja, pero los datos importantes aún se utilizarán en el sistema.)

Los datos que fluyen dentro de un procesamiento suponen un cambio; por lo tanto, el flujo de datos fuera de las fronteras se llama diferentes al que está dentro de las mismas.

En la mayoría de los casos, los nombres del flujo de datos deben aclarar la relación entre los múltiples flujos de datos que entran y salen de los procesos

2.3.15 NOMENCLATURA DE LOS PROCESOS

A todos los procesos deberían asignárseles nombres que le dijeran al lector algo específico sobre la naturaleza de las actividades del mismo. Los nombres CONTROL DE INVENTARIO, COMPRAS Y VENTAS son demasiado generales como para ser útiles. Es mucho mejor decir específicamente que el proceso es AJUSTAR LA CANTIDAD A MANO, PREPARAR LA ÓRDEN DE COMPRA o EDITAR EL PEDIDO DE VENTA para describir el proceso.

El sistema de numeración introducido antes identifica más aún los procesos, en especialmente niveles de detalle. El diagrama de más alto nivel, que define al sistema en su totalidad y que es estudiado, no necesita numerarse. Todos los diagramas de menor nivel deben llevar un número de identificación; por ejemplo, cinco procesos podrían identificarse con números uno, dos, tres, cuatro y cinco. Las extensiones de menor nivel en estos

procesos se asignan con un decimal para indicar que son explicaciones más detalladas de procesos de mayor nivel. Por ejemplo, identifiquense cuatro extensiones de menor nivel del proceso tres como 3.1, 3.2, 3.3 y 3.4. Las extensiones sucesivas añaden más dígitos decimales: 3.1.1, 3.1.1.1, 3.1.1.1.1, etc.

Las figuras 2.3.10 y 2.3.11 utilizan esa convención para explicar los procesos de aprobación de factura y de pago al vendedor, dentro del proceso de cuentas por pagar. Nótese como el número identifica el proceso de mayor nivel que se explica. Cuando se señala un solo diagrama, se sabe el nombre del proceso y el nivel de extensión que se está viendo.

2.4 FLUJO DE DATOS

2.4.1 INTRODUCCIÓN

Como se ha mencionado anteriormente una parte importante de las etapas del desarrollo de sistemas es el análisis. En esta etapa es donde se dimensiona de una manera finita todos los requerimientos del usuario, nos permite ver con más claridad y detalle las partes en las cuales se dividirá el sistema, ayuda a determinar en que tiempo y cuántas personas se requieren para hacerlo, además que es un medio con el cual se le comunica al usuario cómo se va a realizar y si queda alguna duda o hay correcciones es fácil hacerlas en este momento.

A continuación se mostrará el análisis del sistema realizado:

En el primer diagrama vemos el nivel más alto del Sistema de Información para Operaciones Bancarias, donde tenemos dos grandes módulos, el de información (front end) y el del servidor central (main frame). El módulo de información es el que se encarga de presentar al cliente de la sucursal todos los servicios que le puede otorgar el banco, así como el poder realizar operaciones de pago, cobro, transferencias, etc. El módulo del servidor recibe todas las operaciones que manda el módulo de información, decodifica y decide que transacción se ejecutará; la cual extrae o actualiza las bases de datos que se encuentran en este módulo.

En este análisis se verá sólo el módulo de información. Figura 2.4.

El módulo de información se divide en tres módulos de nivel alto, llamados de:

- **Operaciones Bancarias (1) Figura 2.4.1**
- **Productos (2) Figura 2.4.2**
- **Servicios (3) Figura 2.4.3**

El módulo de Operaciones Bancarias (1) se encargará de realizar:

- **Transferencia vista a pagares (1.1) Figura 2.4.1.1**
- **Pago de servicios (1.2) Figura 2.4.1.2**
- **Operaciones financieras (1.3) Figura 2.4.1.3**
- **Cambio de clave secreta (1.4) Figura 2.4.1.4**

- Consultas (1.5) Figura 2.4.1.5

El módulo de Productos (2) se encargará de dar información acerca de los productos que el banco ofrece en ese momento a sus clientes, como información acerca de:

- Crédito (2.1) Figura 2.4.2.1
- Cuenta de Cheques (2.2) Figura 2.4.2.2
- UDIS (2.3) Figura 2.4.2.3
- Tarjeta de Inversión Única (2.4) Figura 2.4.2.4
- Cuenta Única (2.5) Figura 2.4.2.5

El módulo de Servicios (3) proporcionará al cliente la opción de obtener información acerca de los servicios que se ofrecen al cliente, como:

- Electrobanca (3.1) Figura 2.4.3.1
- Telebanca (3.2) Figura 2.4.3.2
- Multicaja (3.3) Figura 2.4.3.3
- Caja Divisa (3.4) Figura 2.4.3.4

El módulo (1.1) Transferencia vista a pagarés, comprará pagarés haciendo cargo a las cuentas de cheques del cliente, además se emitirá un recibo con el estado de la transacción realizada.

El módulo (1.2) Pago de Servicios realizará el pago del recibo de luz o teléfono del cliente, con tan sólo mostrar el código de barras del recibo al lector óptico éste tendrá los

datos necesarios para realizar la operación, el cliente podrá escoger a que cuenta se le hará el cargo del importe correspondiente.

El módulo de Operaciones financieras (1.3) hará operaciones de compra-venta de títulos tomando en cuenta el tipo de fondo y el contrato con el cual el cliente desea realizar su operación.

El módulo de Cambio de clave secreta (1.4) cambiará a gusto del cliente la clave de acceso al sistema.

El módulo de Consulta (1.5) podrá obtener el saldo de:

- cuentas de cheques(1.5.1)
- cuentas únicas (1.5.2)
- tarjetas de inversión (1.5.3)
- tarjetas de ahorro (1.5.4)
- tarjetas de crédito (1.5.5)

El módulo de Crédito (2.1) dará información de los diferentes créditos que otorga el banco, como:

- automotriz (2.1.1)
- tarjeta de crédito (2.1.2)
- hipotecario (2.1.3)
- personal (2.1.4)

El módulo de Cuenta de Cheques (2.2) dará información de las cuentas de cheques que ofrece el banco, así como sus modalidades.

El módulo de UDIS (2.3) dará información del producto de UDIS, tocando los siguientes puntos:

- Definición y renovación (2.3.1)
- Monto y plazos (2.3.2)
- Requisitos (2.3.3)
- Régimen fiscal (2.3.4)
- Productos de inversión (2.3.5)
- Valor de las UDIS (2.3.6)

El módulo de Cuenta única (2.4) dará información del producto de cuenta única, mostrando así:

- atributos y oportunidades (2.4.1)
- requisitos (2.4.2)

El módulo de Tarjeta de inversión (2.5) única dará información de la tarjeta de inversión mostrando:

- Atributos y Oportunidades (2.5.1)
- Requisitos (2.5.2)

2.4.2 DIAGRAMA DE FLUJO DE DATOS

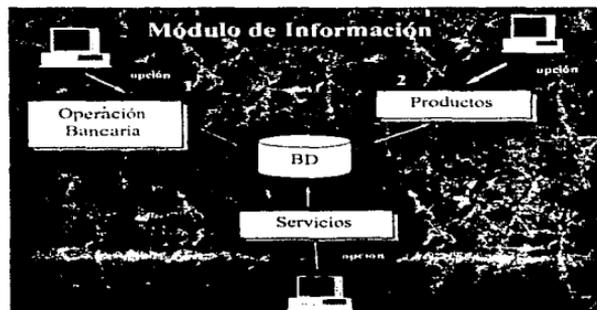


Figura 2.4 Módulo de Información



Figura 2.4.1 Operaciones Bancarias

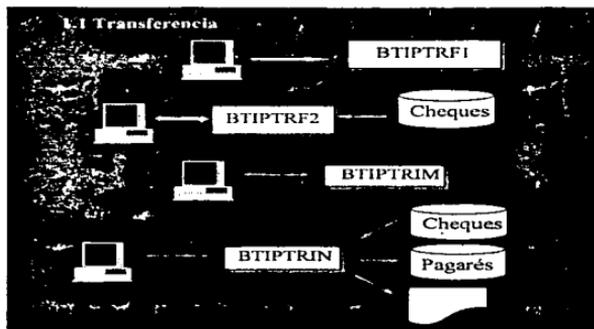


Figura 2.4.1.1 Transferencia

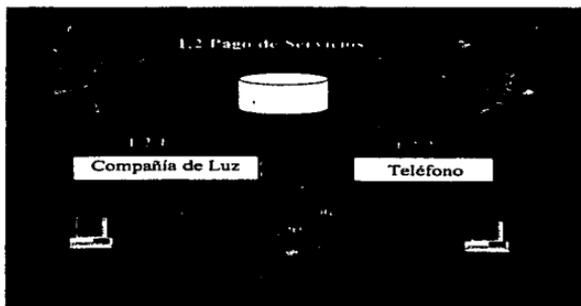


Figura 2.4.1.2 Pago de Servicios



Figura 2.4.1.3 Operaciones Financieras

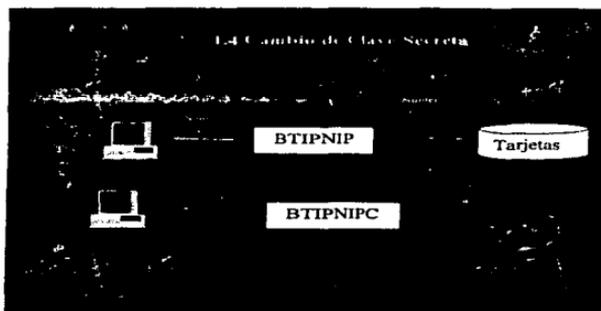


Figura 2.4.1.4 Cambio de Clase Secreta

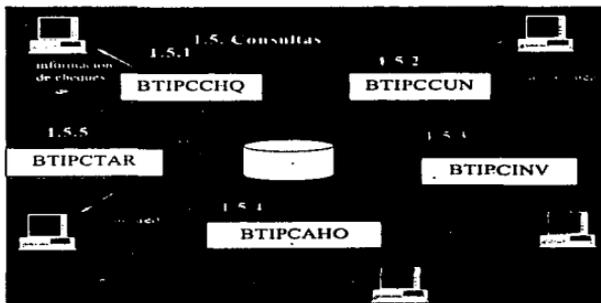


Figura 2.4.1.5 Consultas

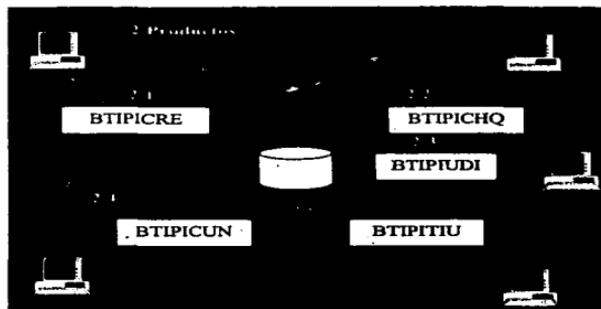


Figura 2.4.2 Productos

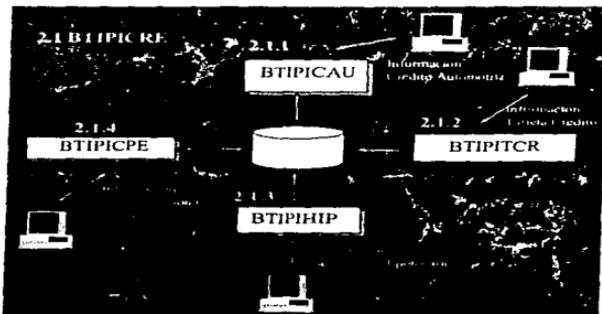


Figura 2.4.2.1 Información sobre Créditos



Figura 2.4.2.2 Información sobre Cuenta de Cheques

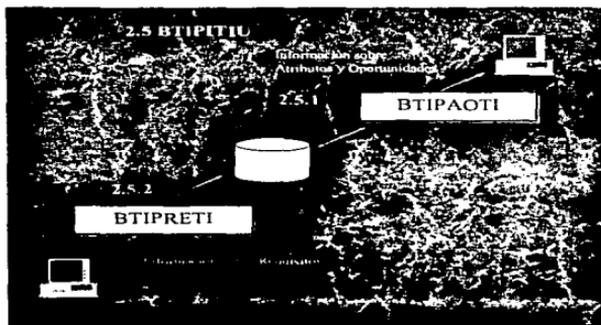


Figura 2.4.2.5 Información sobre Tarjeta de Inversión

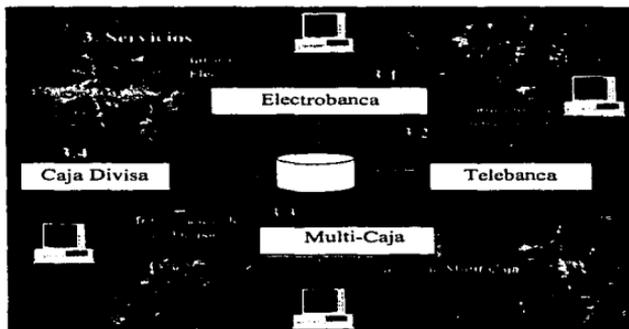


Figura 2.4.3 Servicios

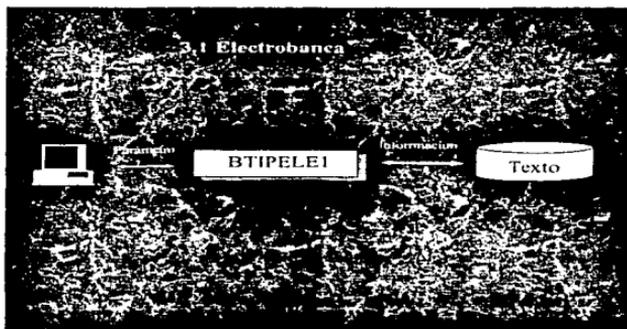


Figura 2.4.3.1 Información sobre Electrobanca

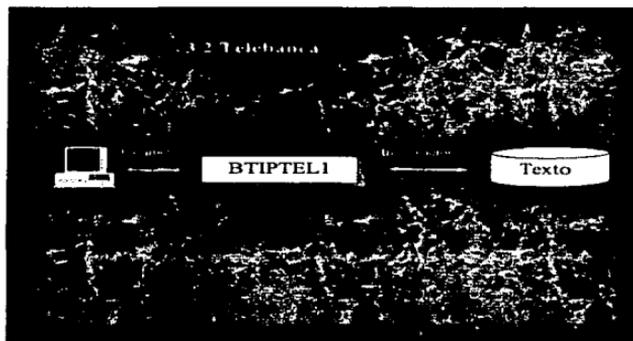


Figura 2.4.3.2 Información sobre Telebanca

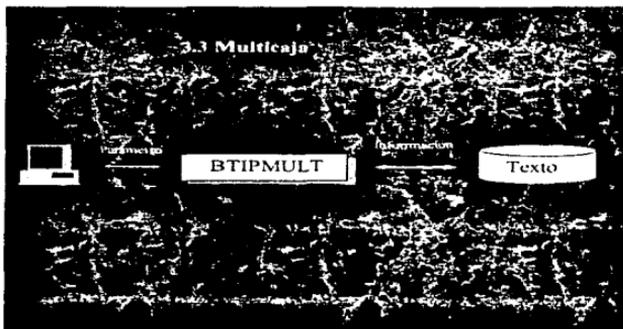


Figura 2.4.3.3 Información sobre Multicaja

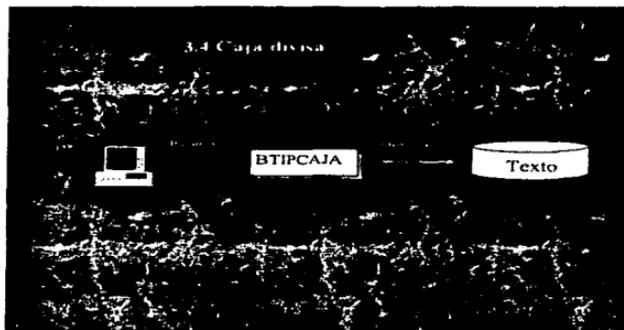


Figura 2.4.3.4 Información sobre Caja divisa

Capítulo 3

Diseño

3.1 ESPECIFICACIONES DE LOS ELEMENTOS LÓGICOS DEL DISEÑO

El diseño de sistemas tiene dos etapas: el diseño lógico y la construcción física del sistema. Cuando el analista formula el diseño lógico, escribe las especificaciones detalladas del nuevo sistema, es decir aquellas que describen sus características: salidas, entradas, archivos y bases de datos y los procedimientos, todo en una forma que satisfaga los requerimientos del proyecto. El conjunto formado por todas estas características recibe el nombre de especificaciones del diseño del sistema.

El diseño lógico de un sistema de información es similar al proyecto de ingeniería de un automóvil: muestra las características más sobresalientes (como el motor, la transmisión y el espacio para los pasajeros) y la relación que guarda entre sí (dónde se conectan los componentes unos a otros o cuál es la separación que existe entre las puertas). Los reportes y salidas generadas por el analista son similares a los componentes de diseño del ingeniero. Los procedimientos y datos se enlazan entre sí para producir un sistema que trabaja.

Al diseñar un sistema de inventarios, por ejemplo, las especificaciones del mismo incluyen definiciones de reportes y pantallas de presentación que describen las existencias disponibles, el abastecimiento y retiro de artículos, y el resumen de transacciones realizadas durante, por ejemplo, un mes de operación. El diseño lógico también especifica los formatos de entrada y la distribución de la salida en pantalla para todas las transacciones y archivos que son necesarios para dar mantenimiento a los datos del inventario, a los detalles de las transacciones y a los datos de los proveedores. Las especificaciones de procedimientos describen los métodos utilizados para ingresar datos en el sistema, copiar archivos y detectar problemas, si éstos se presentaran.

La construcción física, que es la siguiente actividad después del diseño lógico, produce el software, los archivos y un sistema que funciona. Las especificaciones de diseño indican a los programadores lo que el sistema debe hacer. A su vez, los programadores escriben programas que aceptan la entrada proporcionada por los usuarios, procesan los datos, producen los reportes y guardan los datos en los archivos.

El diseño físico para el sistema de inventarios ya mencionados, está formado por instrucciones de programa, escritas en un lenguaje de programación. Estos pasos revisan los registros de mercancía en existencia utilizando para ello los datos asentados en la transacción, imprimen los reportes y guardan los datos. El analista especifica los algoritmos necesarios para cambiar las cantidades de mercancía en las instrucciones necesarias del programa para calcular los cambios y producir los resultados.

3.2 DISEÑO ESTRUCTURADO

Una de las herramientas principales del diseño estructurado es la carta de estructura, la cual muestra la partición del sistema en módulos y la relación jerárquica entre éstos. Además muestra los flujos de datos y control entre los módulos

Una carta de estructura cuenta con los siguientes elementos:

1. Un rectángulo con un nombre inscrito para indicar un módulo. Fig. 3.2.1
2. Líneas que indican la liga entre módulos (llamadas a módulos). Fig. 3.2.2
3. Flechas que indican el flujo de datos y de control respectivamente (comunicación entre módulos). Fig. 3.2.3
4. Un módulo representado como:



significa que es un módulo predefinido.

5. El nombre del módulo debe de resumir los nombres de sus subordinados inmediatos o resumir su función y las funciones de sus subordinados inmediatos.

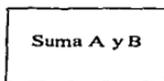


Figura 3.2.1 Módulo

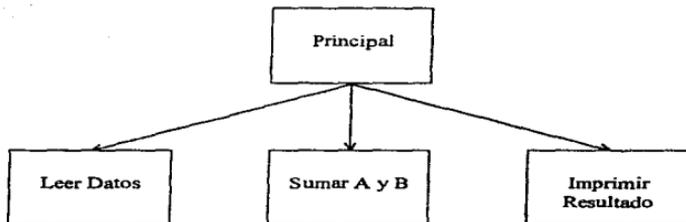


Figura 3.2.2 Ligas entre módulos

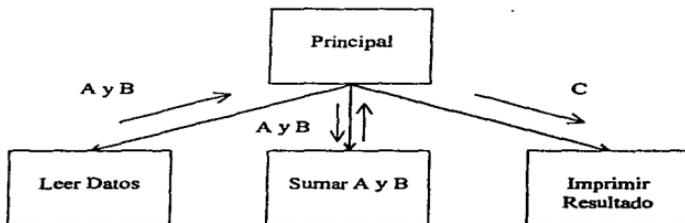
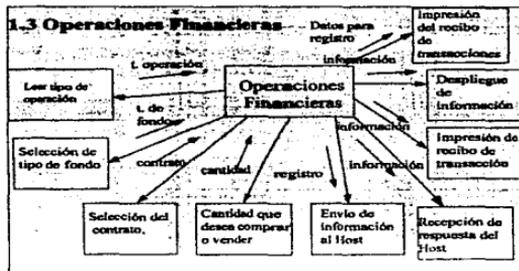
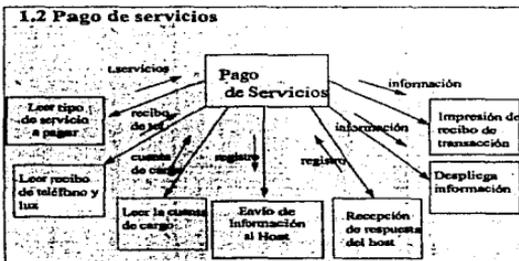
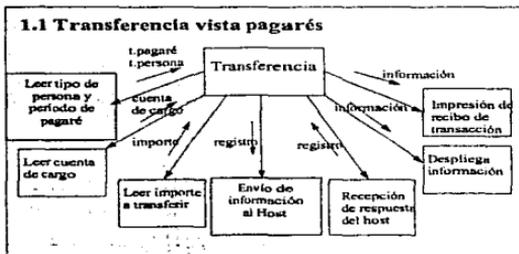
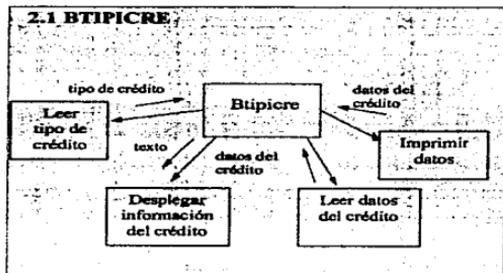
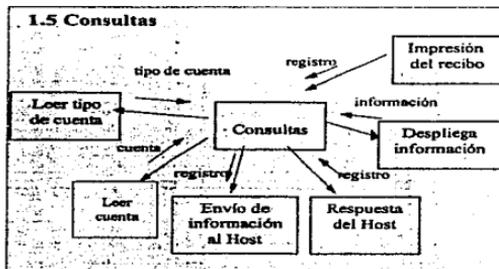


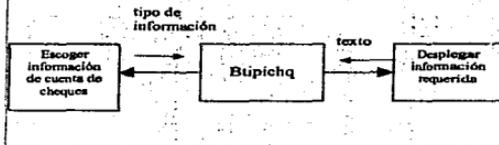
Figura 3.2.3 Comunicación entre módulos

A continuación se muestra la carta estructurada del sistema de información que se está tratando, que se ha generado a través del diagrama de flujo del capítulo anterior.

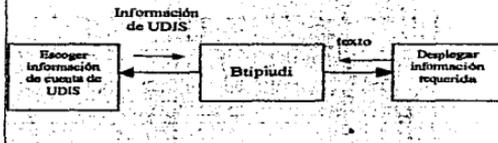




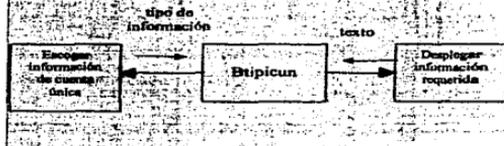
2.2 BTIPICHO



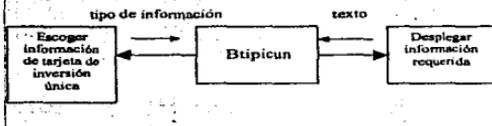
2.3 BTIPIUDI



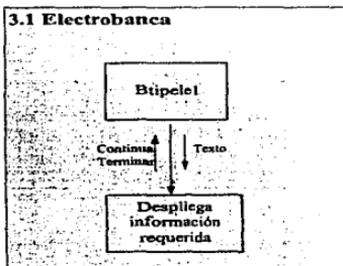
2.4 BTIPICUN



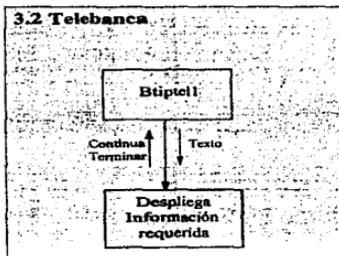
2.5 BTIPITIU



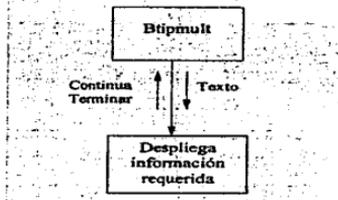
3.1 Electrobanca



3.2 Telebanca



3.3 Multicaja



3.4 Caja Divisa



3.3 PROPORCIONAR ESPECIFICACIONES DETALLADAS PARA EL DESARROLLO DE SOFTWARE

Al igual que las características de un sistema de información, el software también debe ser diseñado con cuidado. El diseño de sistemas incluye la formulación de las especificaciones de software.

Estas especificaciones establecen las funciones de entrada, salida y procesamientos, así como los algoritmos necesarios para efectuarlas. Los módulos de software junto con las rutinas, se enfocan sobre lo que cada función realiza; asimismo, se especifican los procedimientos necesarios para llevar a cabo dichas funciones. La selección de lenguajes de programación, paquetes de software y utilería se efectúa durante el proceso de diseño lógico y las recomendaciones se incluyen como parte de las especificaciones del software.

A continuación se mostrarán algunas de las especificaciones técnicas para realizar el desarrollo del sistema:

1.3

Objetivo: Seleccionar el tipo de persona y el plazo de los pagars.

Programa: BTIPTRF1

- 1 Se muestra la pantalla BTIPTRF1
- 2 Dependiendo de la opción solicitada (Persona Física o Persona Moral) identificar el tipo de persona.
- 3 Dependiendo del periodo de pago de los pagars (Pagará a 28 días, 91 días, 182 días, 370 días) identificar el tipo de pagars.
- 4 Los botones de selección quedarán encendidos para indicar que ya fueron activados.
- 5 Se forma una cadena de caracteres dependiendo de las opciones anteriores, para que lo tomen los siguientes programas.

Objetivo: Obtener las cuentas de cheques asociadas a la tarjeta para que el cliente seleccione una.

Programa: BTIPTRF2

- 1 Mostrar pantalla BTIPTRF2
- 2 Obtener el número de tarjeta de la variable NU1, NUMTAR.
- 3 Generar la cadena de caracteres para enviar la solicitud de las cuentas asociadas a la tarjeta.
- 4 Enviar la cadena de caracteres al host.
- 5 Recibir la cadena con la respuesta del host.
- 6 Desplegar las cuentas asociadas a la tarjeta.
7. El cliente seleccionará de que cuenta de cheques quiere hacer su transferencia

Objetivo: Que el cliente digite la cantidad a transferir

Programa: BTIPTRIM

- 1 Mostrar la pantalla BTIPTRIM
- 2 Recibir la cantidad verificando que no sea igual a 0.00

Objetivo: Desplegar los datos proporcionados por el cliente y si el cliente considera que son correctos efectuar la transacción

Programa: BTIPTRIN

- 1 Desplegar la pantalla BTIPTRIN.
- 2 Desplegar la cantidad y el número de días.
- 3 El cliente dará confirmación. (Se recibe información de la pantalla).
- 4 Si la confirmación fue exitosa se envía la transacción al host.

	LONGITUD	DESCRIPCION
5121	4	CLAVE DE OPERACION
E	1	SEPARADOR
	1	SEPARADOR
ASC1(01)	1	SEPARADOR
=	1	SEPARADOR
101	3	CLAVE DEL PRODUCTO
*	1	SEPARADOR
9999999	7	NUMERO DE CUENTA
=	1	SEPARADOR
00009999999999	15	IMPORTE
E	1	SEPARADOR

5 Se lee del host la cadena de caracteres que contiene la fecha de vencimiento, folio de la transacción, la tasa y el nombre del cliente.

	POSICION	DESCRIPCION
XXXX	1-4	ENCABEZADO
I	5	CODIGO DE ERROR
MULTIBANCO COMERMEX	6-70	IDENTIFICACION DEL BANCO
121	71-73	NUMERO DE DIAS
000000000999999999	83-101	IMPORTE ENTERO
99	103-104	IMPORTE DECIMAL
12/12/96	127-134	FECHA DE VENCIMIENTO
000098	140-145	FOLIO
85	176-177	TASA DE INTERES ENTERO
99	179-180	TASA DE INTERES DECIMAL
JAVIER GONZALEZ	190-224	TITULAR

6 Se imprime un reporte con las siguientes características:

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21

1 2 3 4
123456789012345678901234567890
SISTEMA DE AUTOSERVICIO
INVERLAT
RECIBO TRANSFERENCIA VISTA-PAGARES
DD-MM-AAAA HH:MM:SS
FOLIO 9999
IMPORTE 3999,999,999.99
FECHA DE VENCIMIENTO XXXXXXXXXXXXXXXX
TASA DE RENDIMIENTO 99.99%
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

7 Si ocurre que la confirmación no es exitosa se despliega la pantalla BTIPMOPB

1.3

Objetivo: Seleccionar la compra o venta de títulos.

Programa: BTIPFIN1.

1. Mostrar pantalla BTIPFIN1.
2. Obtener dato de la pantalla.

Objetivo: Seleccionar el tipo de fondo a comprar y a vender.

Programa: BTIPFIN2.

1. Mostrar pantalla BTIPFIN2
2. Obtener dato de la pantalla

Objetivo: Obtener los números de contrato asociados a una tarjeta.

Programa: BTIPCUTI

1. Mostrar pantalla BTIPCUTI
2. Obtener los números de contratos asociados a la cuenta que se encuentra en la variable NU_NUMTAR
3. Generar la cadena de caracteres para enviar la solicitud de las cuentas asociadas a la tarjeta.
4. Enviar la cadena de caracteres al host.
5. Recibir la cadena con respuesta del host.
6. Desplegar los contratos asociados a la tarjeta
7. El cliente seleccionará de que contrato quiere hacer su compra o venta.
8. Formar cadena con la información de la pantalla
9. Enviar al host la cadena.
10. Recibir cadena de caracteres de respuesta.
11. Si la transacción fue exitosa llamar pantalla BTIPTICV
12. Si la transacción no fue exitosa llamar pantalla BTIPTERR y mostrar el mensaje de retorno al host.

Objetivo: Desplegar el resultado de la compra-venta de contratos

Programa: BTIPTICV

1. Mostrar pantalla BTIPTICV
2. Desplegar la fecha, número de títulos comprados o vendidos, la cantidad de la operación, la cuenta de cheques, el folio y el titular.
3. Imprimir el recibo de compra o venta

	1	2	3	4
1	1234567890	1234567890	1234567890	1234567890
2	SISTEMA DE AUTOSERVICIO			
3	INVERLAT			
4	COMPRA DE TITULOS			
5	DD-MM-AAAA		HH:MM:SS	
6	NUMERO DE TITULOS COMPRADOS XXXXXXXXXXXX			
7	OPERACION VALUADA \$999,999,999.99			
8	CARGO A CTA CHEQUES XXXXXXXXXXXXXXXX			
9	FOLIO 9999			
1				
2				
3				
4				
5				
6				
7				
8				
9				
20				

3.4 AJUSTARSE A LOS ESTÁNDARES DE DISEÑO

Como puede observarse, los objetivos del diseño de sistemas son muy amplios y afectan aspectos tanto de la aplicación como de la organización en la que será utilizado el sistema. Por consecuencia, no debe sorprender el hecho de que los grupos de sistemas de información mejor manejados, también mantengan estándares para el desarrollo de sistemas. Las especificaciones de diseño se establecen dentro del marco fijado por los estándares.

Los siguientes son ejemplo de áreas incluidas en el diseño de estándares:

* ESTÁNDARES PARA DATOS

Lineamientos para asignar nombre a los datos y especificar su longitud y tipo. Estos lineamientos son utilizados por todas las aplicaciones desarrolladas por el grupo de sistemas de información. Con frecuencia están contenidas en el diccionario de datos.

A continuación se muestra el diccionario de datos, los cuales tienen los siguientes estándares:

IMPORTE	IM-
CLAVE	CV-
NOMBRE	NB-
NUMERO	NU-
LOG	LG-

FECHA FH-
 TIEMPO TM-
 SWITCH SW-
 CONTADOR CT-
 STRING ST-
 ERROR ER

VARIABLES	TIPO	LONGITUD	DESCRIPCIÓN
CT_DIGITO	Integer		Contador de teclas oprimidas
CT_PROX_TECLA	Integer		Lleva la cuenta de cuantas teclas se han oprimido
CV_TFONDO	Integer		Tipo de fondo
CV_TIPOPE	Integer		Tipo de operación
CV_WS_DATSEN	string	variable	Clave de operación a realizar es el host
ER_IMPRESORA	const		Código de error de la impresora
FH_IMP	string	variable	Fecha de impresión del recibo
IM_CAPPAG	double		Capacidad de pago
IM_CAPPAG1	double		Capacidad de pago 1
IM_CAPPAG2	double		Capacidad de pago 2
IM_CAPPAG3	double		Capacidad de pago 3
IM_ENGINI	double		Enganche inicial
IM_FACAJ CASA	const		Factor ajuste casa
IM_FACFINCASA	const		Factor financiamiento casa
IM_FININV	double		Financiamiento de la inversión
IM_GASTAUX	double		Auxiliar para gasto de crédito
IM_GASTOCRE	double		Gasto de crédito
IM_GASTONOT	const		Gasto Notario
IM_GMES	double		Importe de gastos mensuales
IM_IMPORTE	double		Importe a pagar en un pago de servicios
IM_INGMENNENET	double		Guarda el ingreso mensual neto
IM_INGMES	double		Importe del ingreso mensual
IM_MENINI	double		Mensualidad Inicial
IM_PAGMILCASA	const		Factor de pago casa
IM_SALDO	string	variable	Saldo en cuenta de cheques
IM_TOTALTIT	double		Número total de títulos a comprar/vender
IM_VALINMU	double		Valor del inmueble
LG_ACCION	string	50	Indica que acción se realiza para registrar en la bitácora
NU_CANT	double		Número de títulos a comprar
NU_CHEQO	string	7	Número de cuenta de cheques
NU_CTONDO	string	variable	Número de cuenta fondo
NU_CTAUX	string	variable	Número de cuenta auxiliar

NU_FACTOR 1	constant		Factor constante 1
NU_FACTOR 2	constant		Factor constante 2
NU_FACTOR 3	constant		Factor constante 3
NU_FOLIO	string	4	Número de Folio
NU_INVERSION	double		Número de cuenta de inversión
NU_NUMTAR	string	20	Número de tarjeta
NU_NUMTIT	Integer		Número de títulos a comprar o vender
NU_PAGARES	Integer		Indica los días de vencimiento de un pagaré
NU_PERSONA	Integer		Indica si es persona física o moral
NU_PLAZHIP	Integer		Años de plazo de la hipoteca
NU_SUCURSAL	string	variable	Número de sucursal
NU_TASAS	string	variable	Tasa de interés
NU_TIPOCUEN	Integer		Indica el tipo de cuenta seleccionado
NU_TITFOLIO	Integer		Número de folio para títulos
NU_VALNUM	double		Auxiliar para guardar el valor numerico de string
ST_AUXILIAR	string	variable	Auxiliar para formato de cantidades
ST_BANCO	constant		Nombre del banco para impresión de reportes
ST_CADENA	string	variable	Guarda cada una de las teclas oprimidas
ST_CCOMP	string	variable	Forma el string que se envia al host
ST_CODIGO	string	variable	Código de barras
ST_CORTE	string	variable	Envia la instrucción de corte de papel
ST_CTAHORRO	arreglo de 5 strings	6	Cuentas de ahorro asociadas a un cliente
ST_CTACREDIT	arreglo de 5 strings	16	Cuentas de crédito asociadas a un cliente
ST_CTACHEQUE	arreglo de 5 strings	7	Cuentas de cheques asociadas a un cliente
ST_CTAINVÉR	arreglo de 5 strings	14	Cuentas de inversión asociadas a un cliente
ST_CTAUNICA	arreglo de 5 strings	7	Cuentas unicas asociadas a un cliente
ST_CUENTA	string	variable	Almacena un número de cuenta
ST_CVEPRÓD	string	variable	Indica las características de la transferencia a realizar
ST_DS	string	variable	Guarda la respuesta del host
ST_DATAS	string	variable	Información leída desde el puerto de comunicación
ST_DATOS	string	variable	Recibe o envia los datos al host
ST_FINAL	string	variable	Mensaje final leído desde el puerto de comunicación
ST_INFOHOST	string	274	Información que envia el host
ST_LEENIP	string	variable	Lee el nip
ST_NIP	string	4	NIP
ST_NIPPAUX	string	variable	Guarda nip para hacer cambio
ST_NUEVO_NIP	string	variable	Guarda el nuevo nip
ST_PLAZA	constant		Nombre de la plaza
ST_TITULO	constant		Título de recibo
ST_TITULOS	string	130	Guarda información de títulos
SW_CAMBIONIP	Integer		Indica secuencia de consulta o cambio de NIP
SW_CARGO	Integer		Indica si se hará un cargo o abono
SW_CLIENTE	Integer		Switch par saber si es cliente o no
SW_COMPRA	Integer		Indica si es una compra o venta de títulos
SW_CORRECTO	Integer		Indica si el nip fue correcto

SW_HSALDO	string	Variable	Indica si hay saldo
SW_IMPRESORA	boolean		Indica si funciona la impresora
SW_LUZ	Integer		Indica si se va a pagar luz o teléfono
SW_NIP2	Integer		Indica que se da segundo nip para verificar
SW_NONIP	Integer		Indica que no hay cambio de NIP
SW_NUEVO_NIP	Integer		Indica si va a cambiar el nuevo nip
SW_OPCIONIG	Integer		Opción de ingreso mensual
SW_RESULT	Integer		Indica si la lectura al host fue efectuada
TM_IMP	String	Variable	Hora de impresión del recibo
TM_TIEMPO	Integer		Tiempo transcurrido

Para programas y funciones se tiene el siguiente estándar:

B banco

TI transferencia de información

F o P si se trata de una función o programa.

PROGRAMAS O FUNCIONES	DESCRIPCIÓN
BTIFCAHO	Consulta de saldo en cuenta de ahorros
BTIFCCHO	Consulta de saldo en cuenta de cheques
BTIFCINV	Consulta de saldo en cuenta de inversión
BTIFCOM1	Inicializa puerto de comunicaciones
BTIFCOM2	Elimina caracteres de control en código de barras
BTIFCOVE	Realiza operación de compra o venta
BTIFCTAR	Consulta de saldo en tarjeta de crédito
BTIFFOLI	Genera el número de folio de cada transacción
BTIFGUAK	Guarda el nuevo NIP
BTIFHSAL	Verifica que haya saldo para compra o venta
BTIFIMPH	Imprimir crédito hipotecario
BTIFIMPR	Impresión recibo transferencia
BTIFINVE	Realiza operación de inversión
BTIFSUCH	Suma cantidad vendida a cheques
BTIFVERI	Función que verifica si es correcto el NIP
BTIPAOC2	Atributos y oportunidades de cuenta única 2
BTIPAOCU	Atributos y oportunidades de cuenta única
BTIPAOTI	Atributos y oportunidades de tarjeta de inversión única
BTIPAUT2	Información sobre crédito automotriz 2
BTIPAUT3	Información de crédito automotriz 3
BTIPAYHI	Ayuda crédito hipotecario
BTIPBITA	Escribe el log
BTIPCAHO	Despliega consulta de saldos de cuenta de ahorro
BTIPCAJA	Información sobre caja divisas
BTIPCBAR	Lectura del código de barras
BTIPCCUN	Despliega consulta de saldos de cuenta única
BTIPCCHO	Despliega consulta de saldos de cuenta de cheques

BTIPCNV	Despliega consulta de saldos de cuenta de inversión
BTIPCLIE	Programa para saber si es cliente
BTIPCLNO	Menú principal de información
BTIPCLSI	Menú principal para clientes
BTIPCOCR	Consulta tarjeta de crédito
BTIPCOIN	Despliega las cuentas asociadas a cuenta única
BTIPCONS	Menú de consultas de saldo
BTIPCOTA	Consulta de tasas
BTIPCREC	Indica que deposite recibo
BTIPCUEN	Despliega las cuentas asociadas
BTIPCUTI	Selecciona cuenta para compra o venta
BTIPDEUD	Definición de UDIS
BTIPELE1	Información sobre Electrobanca 1
BTIPELE2	Información sobre Electrobanca 2
BTIPFIN1	Operaciones Financieras 1 (selección de compra o venta)
BTIPFIN2	Selección del tipo de fondo a compra o venta
BTIPIHP2	Información sobre crédito hipotecario 2
BTIPIBAJ	Indica que su ingreso es bajo
BTIPICAU	Información sobre crédito automotriz
BTIPICPE	Información sobre crédito personal
BTIPICRE	Menú de información sobre créditos
BTIPICUN	Información sobre cuenta única
BTIPICHO	Información sobre cuenta de cheques
BTIPIHIP	Información sobre crédito hipotecario
BTIPIMEN	Cálculo de ingreso mensual para crédito hipotecario
BTIPIMTI	Indicar importe de títulos a compra o venta
BTIPINCA	Cantidad a invertir
BTIPISCU	Selecciona cuenta para pago de servicios
BTIPITCR	Información sobre tarjeta de crédito
BTIPITIU	Información sobre tarjeta de inversión única
BTIPIUDI	Información sobre UDIS
BTIPIETA	Programa que lee tarjeta
BTIPMECR	Menú de opciones de crédito hipotecario
BTIPMNIP	Indica que estuvo mal el NIP
BTIPMOCH	Modalidades de cuentas de cheques
BTIPMOPB	Menú operaciones bancarias
BTIPMOPL	Monto y plazo de crédito hipotecario
BTIPMOUD	Montos de UDIS
BTIPMULT	Información sobre Multicaja
BTIPNINF	No se encontró información
BTIPNIP	Pregunta el NIP
BTIPNIPC	Indica que el NIP fue correcto
BTIPNIPD	Indica que los nips fueron diferentes
BTIPPAGA	Pago de servicios
BTIPPER2	Información sobre crédito personal 2
BTIPPER3	Información de crédito personal 3
BTIPPER4	Información de crédito personal 4
BTIPPRIN	Programa inicial
BTIPPROD	Menú de productos
BTIPPRON	Despliega promoción
BTIPPRUD	Productos de Inversión en UDIS

BTIPPCU	Selecciona cuenta para pago de servicios
BTIPPSER	Pago de servicios
BTIPRECU	Requisitos de cuenta única
BTIPRETI	Requisitos de tarjeta de inversión única
BTIPREUD	Requisitos para UDIS
BTIPRFUD	Regimen fiscal UDIS
BTIPSERV	Menú de servicios
BTIPSUES	Sugerir crédito especial
BTIPSULI	Sugerir crédito liquidez
BTIPTCCH	Información sobre tipos de cuentas de cheques
BTIPTEL1	Información sobre Telebanca 1
BTIPTEL2	Información sobre Telebanca 2
BTIPTERR	Error en la transacción
BTIPTICV	Compra o venta de títulos
BTIPTINV	Indica que hubo error en la tarjeta
BTIPTRF1	Transferencia 1
BTIPTRF2	Seleccionar la cuenta de cheques para transferencia
BTIPTRIM	Indica el importe de la transferencia
BTIPTRIN	Información de la transferencia
BTIPVAUD	Valor de UDIS
BTIPVIDE	Pantalla que muestra el video

*ESTÁNDARES DE CODIFICACIÓN

Abreviaturas y designaciones formales para describir actividades y entidades dentro de la organización (por ejemplo, clasificación de clientes y tipos de transacciones).

*ESTÁNDARES ESTRUCTURALES

Lineamientos sobre cómo estructurar el software y el sistema. Políticas para dividir el software en módulos, para la codificación estructurada y la relación existente entre los componentes del sistema. Pueden incluir estándares sobre la longitud del programa y lineamientos para volver a utilizar los módulos de software.

***ESTÁNDARES DE DOCUMENTACIÓN**

Descripciones de las características del diseño de sistemas, de la relación entre componentes y de las características de operación que puede ser revisadas para conocer los detalles de la aplicación.

***¿QUÉ CARACTERÍSTICAS SON LAS QUE SE DEBE DISEÑAR?**

Las especificaciones de diseño describen las características del sistema, los componentes o elementos del sistema y la forma en que éstos aparecerán ante los usuarios. Para muchos usuarios, el éxito de un sistema está relacionado con la creencia que tenga sobre sí el sistema tiene las características adecuadas.

Esta sección describe las características que debe diseñar el analista de sistemas. Pero antes de considerarlas, es conveniente primero aclarar qué elementos tiene que tomarse en cuenta en las especificaciones formales de diseño.

3.5 DISEÑO DE LA SALIDA

El término salida, como es probable que el lector lo conozca, se refiere a los resultados e información generados por el sistema. Para muchos usuarios finales, la salida es la única razón para el desarrollo del sistema y la base sobre la que ellos evaluarán la utilidad de la aplicación. En la realidad, muchos usuarios no operan el sistema de

información y tampoco ingresan datos en él, pero utilizan la salida generalmente por el sistema.

Cuando diseñan la salida, los analistas deben realizar lo siguientes:

- Determinar qué información presentar.
- Decir si la información será presentada en forma visual, verbal o impresa y seleccionar el medio de salida.
- Disponer la presentación de la información en un formato aceptable.
- Decidir cómo distribuir la salida entre los posibles destinatarios.

La disposición de la información sobre una pantalla o documento impreso se denomina distribución.

Para llevar a cabo las actividades antes mencionadas, se requieren decisiones específicas tales como el empleo de formatos ya impresos cuando se preparan reportes, cuántas líneas planear sobre una página impresa o si se deben emplear gráficas y colores.

El diseño de la salida está especificado en los formularios de distribución, que son hojas que describen la ubicación, características (como longitud y tipo) y formato de los encabezados de columnas y la paginación. Estos elementos son análogos al bosquejo donde el arquitecto indica la localización de cada componente.

A continuación se muestra una salida del sistema en la cual se emplean los estándares establecidos para la presentación de un reporte de esta naturaleza.

```
1 1234567890123456789012345678901234567890
2 SISTEMA DE AUTOSERVICIO
3 INVERLAT
4 COMPRA DE TITULOS
5 DD-MM-AAAA HH:MM:SS
6
7 NUMERO DE TITULOS COMPRADOS XXXXXXXXXXXX
8 OPERACION VALUADA $999,999,999.99
9
10 CARGO A CTA CHEQUES XXXXXXXXXXXXXXXX
1 FOLIO 9999
2
```

3.6 DISEÑO DE LA ENTRADA

Los analistas de sistemas deciden lo siguientes detalles del diseño de entradas:

- 1.- Qué datos ingresan al sistema.
- 2.- Qué medios utilizar.
- 3.- La forma en que se deben disponer o codificar los datos.
- 4.- El diálogo que servirá de guía a los usuarios para dar entrada a los datos.
- 5.- Validación necesaria de datos y transacciones para detectar errores.
- 6.- Métodos para llevar a cabo la validación de las entradas y los pasos a seguir cuando se presentan errores.

Las decisiones de diseño para el manejo de entradas, especifican la forma en que serán aceptados los datos para su procesamiento por computadora. Los analistas deciden si los datos serán proporcionados directamente, quizá a través de una estación de trabajo, o por el uso de documentos, como talones de ventas, cheques bancarios o facturas, dando los datos que a su vez son transferidos hacia la computadora para su procesamiento.

El diseño de la entrada también incluye la especificación de los medios por los que tanto los usuarios finales como los operadores darán instrucciones al sistema sobre las acciones que debe emprender. Por ejemplo, un usuario que interactúa con el sistema por medio de una estación de trabajo, tiene que ser capaz de indicarle al sistema ya sea que acepte una entrada, genere un reporte o termine el procesamiento.

A continuación se mostrarán algunas de las pantallas de entrada de información diseñadas para el sistema:

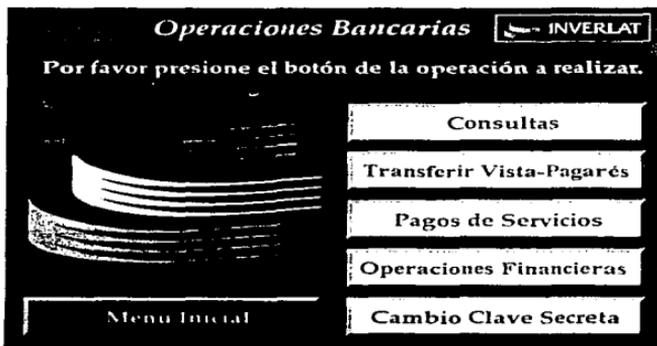


Figura 3.6.1 Pantalla para seleccionar la operación bancaria a realizar

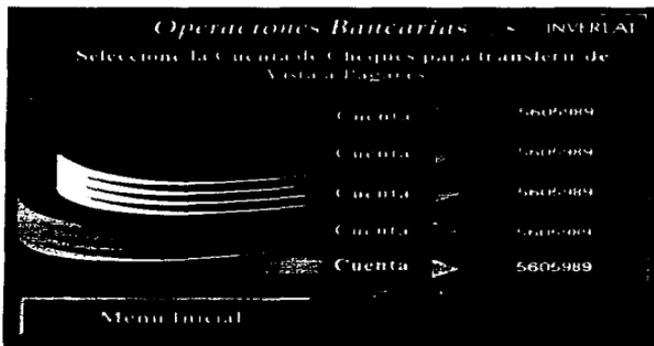


Figura 3.6.2 Pantalla para seleccionar la cuenta de cheques para realizar una transferencia



Figura 3.6.3 Pantalla para indicar el importe de una transferencia

3.7 DISEÑO DE CONTROLES

Los analistas de sistemas también deben anticipar los errores que se cometerán al ingresar los datos en el sistema o al solicitar la ejecución de ciertas funciones. Algunos errores no tienen importancia ni consecuencias, pero otros pueden ser tan serios que ocasionarían el borrado de datos o el uso inapropiado del sistema. Aunque exista sólo la más mínima probabilidad de cometer un error serio, un buen diseño de sistema de información ofrecerá los medios para detectar y manejar el error.

Los controles de entrada proporcionan medios para:

- 1) asegurar que sólo los usuarios autorizados tengan acceso al sistema apoyándose de una tarjeta de encriptación DES
- 2) garantizar que las transacciones sean aceptables verificando los formatos para que no sean rechazadas por el host
- 3) validar los datos para comprobar su exactitud
- 4) determinar si se han omitido datos que son necesarios

Capítulo 4

Desarrollo

4.1 INTRODUCCIÓN

En este capítulo se verán las características del equipo con que se trabajó y cómo fue instalado para el funcionamiento total del sistema. Además se explicará cómo se desarrollaron los módulos del sistema y cada una de las particularidades de ellos.

Para la implementación del Sistema para Transacciones Bancarias era necesario un lenguaje que nos permitiera una fácil utilización de la pantalla sensible al tacto por el usuario, es decir, que para el manejo de objetos como son botones, campos de despliegue, etiquetas y pantallas, no se requiriera de mucho código de programación. Estas condiciones las cumple un lenguaje orientado a eventos es decir uno que tuviera definidos varios objetos y acciones asociadas a éstos, programadas previamente, a los cuales se les asocia un código para realizar un proceso determinado por el programador.

Cabe señalar que cuando se habla de un objeto no nos referimos a las estructuras de datos que se utilizan en la programación orientada a objetos, ya que en ese caso un objeto tiene reglas definidas de herencia determinada que no son necesarias en este proyecto. Por esto no se utilizó C++ u otro lenguaje orientado a objetos.

Ejemplo: Cuando un usuario oprime en la zona de la pantalla que esta definida con el objeto BOTON1, se dispara el evento MouseDown, y se programa la instrucción que muestra el objeto LINEA1, para indicar que el BOTON1 fue seleccionado. Cuando el usuario retira su dedo de la zona de BOTON1, se realiza el evento MouseUp, al cual se le asocia la instrucción que oculta el objeto LINEA1 y en el evento Click se indica el código de la acción relacionada a la selección de BOTON1.

Para el ejemplo anterior no es necesario definir mediante programación las coordenadas que va a ocupar el botón o la línea, únicamente el programador tiene que dibujarlos en la pantalla con el Mouse. Tampoco es necesario realizar la programación para sensar la interrupción de sistema operativo que indica que se oprimió una parte determinada de la pantalla.

Las opciones más comunes en el mercado para programación orientada a eventos son Power Builder y Visual Basic. Power Builder es el líder mundial pero en México no esta muy difundido y nos inclinamos por Visual Basic 3.0 de Microsoft, por ser un lenguaje que es más conocido y esto representa mayor facilidad para comercializar el sistema en el país.

Otro punto determinante en la elección de Visual Basic fue el hecho de que en el banco ya existían rutinas programadas con anterioridad para establecer la comunicación con el servidor, que el control para reproducir los videos también estaba desarrollado para este

lenguaje y que el banco está cambiando su estándar de programación hacia este lenguaje por su facilidad y versatilidad.

4.2 PROGRAMACIÓN Y DESARROLLO

Una vez realizada la elección del lenguaje, se procedió a programar las consultas de saldo, ya que éstas eran las más sencillas por que no se realiza ninguna modificación a la información que se encuentra en el servidor, como es el caso de transferencias y pago de servicios.

Para efectuar una transacción con el servidor, es necesario formar una cadena de caracteres con una distribución definida previamente por el banco, de tal manera que se debe indicar el código de la operación a realizar, la cuenta a afectar, la cantidad, la sucursal, la plaza, ciertos códigos de control, etc. El servidor nos enviará una cadena que nos indica primeramente si la transacción pudo ser realizada correctamente o una leyenda indicando cual fue el problema. Si la transacción fue exitosa, la cadena contendrá los datos solicitados correspondientes al código de operación indicado, éstos pueden ser número de cuenta, saldo, fecha de corte, cantidades transferidas, etc. Para realizar estas transacciones, más adelante se explicarán los procedimientos para establecer una sesión en la red y para enviar o recibir la información.

Visual Basic trabaja basándose en formas, una forma es una ventana o una caja de diálogo que es parte de una interfaz de aplicación. Las formas tienen propiedades que

determinan aspectos de su apariencia, como posición, tamaño y color, y aspectos de su comportamiento como en donde se podrán cambiar su tamaño. Las formas pueden responder a eventos indicados por el usuario o disparados por el sistema. Por ejemplo, se puede escribir código en el procedimiento del evento Click de una forma que permitirá al usuario cambiar el color de una forma haciendo click sobre ella.

Las formas se guardan en archivos .FRM en los cuales se almacena cada uno de los controles definidos para éstas, sus características y código. Un proyecto en Visual Basic es un conjunto de formas interrelacionadas mediante programación. Los proyectos se guardan en archivos .MAK los cuales contienen los nombres de los archivos de formas a utilizar.

En nuestro caso, como deseábamos que nuestra aplicación fuera altamente atractiva al cliente y no se viera como un típico sistema de Windows, al definir las formas, se eliminaron los bordes, los controles para modificar su tamaño y la barra de título, para lograr esto, se definieron las siguientes propiedades.

BackColor	&H00000000& (negro)	Evita que se vea una línea blanca en la parte inferior e izquierda de la forma ya que las formas y las pantallas tienen una diferencia en tamaño de un píxel en ancho y alto
BorderStyle	None	Elimina los bordes que permiten ajustar el tamaño de la forma y la barra de título
Picture	(Bitmap)	Asocia un archivo .BMP para que sea desplegado en el fondo de la forma
WindowState	2 – Maximized	Permite que la forma ocupe toda la pantalla

Otro motivo por el que se definieron estas características y se eliminaron los controles es para evitar que el público en general pudiera interferir con el funcionamiento de la terminal al tener acceso a Windows, sistema operativo u otra aplicación.

Se pueden definir variables para cada forma, estas variables son locales y pierden su valor al cerrar la forma, si deseamos contar con variables globales, éstas se definen en un módulo independiente que se agrega al proyecto. En nuestro caso se creó el módulo VAR.GLO, el cual contiene variables globales para transferir información entre formas, como es el caso de números de cuentas, importes, banderas, etc. También existen constantes que se definen en este módulo y nos permiten localizarlas con mayor facilidad cuando se presenta un cambio, en este caso se encuentran tasas de interés, nombres de sucursal, nombre del banco, plazos, etc.

PANTALLAS

El banco inicialmente propuso un formato muy sencillo para cada una de las pantallas que se iban a utilizar en el sistema. Como gran parte de la idea del sistema era desarrollar una interfase atractiva al cliente, se vio la necesidad de utilizar un paquete de diseño gráfico para crear pantallas con una calidad mejor a la que nos podía proporcionar Visual Basic, ya que el formato podría verse como un típico sistema de Windows. Las pantallas fueron diseñadas con Corel Draw 4.0 dándoles un colorido, distribución y controles más atractivos a la vista.

Las pantallas debieron transformarse a un formato .BMP de 256 colores sin compresión, de 480 X 640 pixels, que es el formato que utiliza Visual Basic para leer las imágenes. Las pantallas se incorporan a las formas en la propiedad Picture, donde se indica el archivo de donde se debe de obtener la imagen.

Para el manejo de los botones, no se utilizó el objeto botón ya que en este caso, los botones eran parte de la imagen del fondo de la forma. De esta manera para simular la selección de un botón se colocó sobre el área que indica la existencia de un botón, un objeto imagen transparente Este objeto nos permite detectar cuando se realiza un click en el área que simula el botón. Para mostrar que se seleccionó el área que simula el botón cuando se detecta el evento MouseDown se muestra un objeto shape, el cual puede ser una línea en la

parte inferior del botón o un rectángulo alrededor del área del botón, y se esconde cuando se produce el evento MouseUp.

Los menús de Productos y Servicios nos presentan preferentemente información, requisitos y características de cada uno de éstos. Todos estos datos se encuentran en archivos de texto predefinidos, de tal manera que al modificarse alguna información, sólo sea necesario copiar un nuevo archivo a la terminal ya sea a través de la red o por diskette. Estos archivos de texto son presentados en objetos label que se encuentran en la pantalla. Un objeto label es una etiqueta de texto el cual tiene propiedades como tamaño, ancho y color de la letra, y en la propiedad caption se indica el texto a desplegarse.

En algunos casos, debido a la gran saturación de información con que contaba la pantalla y para no cansar al usuario, se colocó un objeto reloj en la forma asociado al despliegue de los textos, de tal manera que cada dos segundos se fuera presentando un párrafo de la información.

MOTHERBOARD

El MotherBoard del equipo es Intel, modelo 486 Classic PCI Serie R. Este modelo es compatible con toda la familia de microprocesadores Intel 486, soportando adicionalmente el procesador OverDrive basado en arquitectura Pentium.

Integrados en el MotherBoard se encuentran los siguientes controladores: disco duro PCI/IDE, unidad de diskette, teclado, mouse y video, el que incluye 512 KB de memoria de video DRAM. Además están integrados 4 MB de memoria RAM.

El MotherBoard provee por medio de un Riser Card tres Slot ISA, en uno de los cuales se puede conectar una tarjeta de red para integrar el equipo a los distintos tipos de redes. Todos los conectores del MotherBoard son estándares; el teclado y el Mouse son del tipo PS/2, los puertos seriales son DB9M, el puerto paralelo es del tipo DB25F, el conector de video es del tipo 15 pines análogo, los conectores para el disco duro y la unidad de diskette son del tipo doble corrida de pines para cable plano.

TARJETA ETHERNET PARA COMUNICACIÓN EN RED

La tarjeta de comunicaciones seleccionada fue la 3Com EtherLink III Parallel Tasking de 16 bits para adaptador Industry Standard Architecture (ISA) por ser una de las tarjetas más comunes en el mercado, su facilidad de instalación, compatibilidad y precio.

Esta tarjeta se presenta en 4 modelos diferentes dependiendo del conector ethernet con que cuenta (RJ45, AUI, BNC). La red en el banco utiliza cableado Ethernet de par trenzado también conocido como 10BASE-T, el cual utiliza conector RJ45, pero como el cableado podía cambiar dependiendo de la sucursal donde se instalara el equipo, se decidió utilizar el modelo COMBO, el cual cuenta con los tres tipos de conectores.

Una vez instalada la tarjeta y conectado el cable de red, es necesario configurarla mediante un diskette proporcionado por el proveedor. Se corren los programas AUTOLINK e INSTALL y en éste seleccionamos la opción Auto Configure 3Com ISA Adapter y esperamos a que aparezca el mensaje de configuración exitosa.

A continuación se deben instalar los drivers de red, para esto se debe seleccionar la opción Network Drivers en el menú principal y ahí indicar el sistema operativo de red que estamos utilizando.

Para establecer la comunicación con el servidor, el banco nos proporcionó el código fuente que utiliza para comunicar estaciones de trabajo en la red, éste se encuentra en el módulo landp.bas, un módulo es un conjunto de subrutinas o procedimientos agrupados en un archivo. Así como el dll alberto.dll, por dll .se entiende Librería Ligada Dinámicamente (Dynamic Link Library) y es un una librería de rutinas cargadas y ligadas en el momento de correr el programa.

Una sesión con el servidor necesita de varios pasos para ser concretada:

Se corre el procedimiento inicializa, el cual establece valores iniciales a las variables a utilizar durante la sesión.

IN_landp es una función que verifica que nuestra terminal está posibilitada para iniciar una sesión (la tarjeta de red y los drivers están funcionando correctamente), establece el protocolo para comunicarnos y regresa un valor indicando el resultado.

La función **Abre_Sesión** establece una sesión con el servidor y verifica que se puedan recibir y enviar datos.

Para enviar información al servidor, se crea la cadena **cprb.ehcqdataad** con caracteres en código EBCDIC y en **cprb.ehcqdata1** se guarda la longitud de esta cadena. Se corre el procedimiento **Send_Host**.

Si deseamos leer información enviada desde el servidor utilizamos la función **Read_Host**, la cual deberá encontrarse dentro de un ciclo **While** que nos indica que el servidor está enviando la información mediante la instrucción **get_status**. La cadena de caracteres **ws_dato** contiene los datos enviados por el servidor y es necesario transformarlos a formato ASCII ya que se encuentran en formato EBCDIC.

Cuando se desea terminar una sesión con el servidor se ejecuta el procedimiento **Realease_Host**.

El procedimiento **Eject_Landp** descarga el protocolo de red en la terminal.

TARJETA PARA REPRODUCCIÓN DE VIDEO

Como era necesario reproducir un video en la terminal mientras no se estuviera utilizando para llamar la atención del cliente, se investigó cuál es el método más conveniente para la reproducción de videos a bajo costo (sin necesidad de instalar un CD ROM o Laser Disc) y con mejor calidad que los actuales formatos como son los .avi en los cuales la animación se puede ver como salta entre cuadro y cuadro, además de ocupar una gran cantidad de espacio en disco duro.

MPEG (Moving Picture Experts Group) Grupo de Expertos en Imágenes en Movimiento se refiere a un tipo de compresión optimizada para datos y video. Analizando los cambios entre cuadros, un "encriptor" MPEG es capaz de reducir el tamaño de la información a razón de 200 a 1. Este eficiente método de compresión permite que verdaderos videos de pantalla total y movimiento total, además de audio con calidad de Compact Disc sean almacenados en un archivo relativamente pequeño.

Una vez que la información ha sido comprimida, debe ser descomprimida antes de poder ser usada. Para esto se seleccionó la tarjeta RealMagic Lite de Sigma Designs por ser la opción más utilizada en el mercado que incorpora hardware para descomprimir información MPEG a una muy alta velocidad, lo que permite a los "flujos" de MPEG reproducirse a una tasa de 150KB por segundo. Los videos comprimidos de movimiento total pueden contar con 128,000 colores de video mostrados a una resolución de 1024x768

a tasas de 24, 25 ó 30 cuadros por segundo, y el audio comprimido puede reproducir a 44.1 kHz, en estéreo y con 16 bits.

La tarjeta se coloca en un slot de expansión ISA de 16 bits y cuenta con un cable que se conecta a la tarjeta de video VGA. Es necesario conectar un terminador proporcionado por el fabricante en la salida para el monitor de la tarjeta VGA y el cable del monitor ahora se conectará a la salida de la tarjeta RealMagic. Así mismo, la tarjeta RealMagic cuenta con una salida donde se pueden conectar las bocinas para la reproducción del sonido.

Para la instalación de los drivers de la tarjeta, el fabricante proporciona un diskette e indica correr un programa para la configuración. Se seleccionó la opción Standard Installation pero como el sistema cuenta con una tarjeta de comunicaciones, fue necesario utilizar la opción Custom Installation por que se presentaron problemas de colisiones en las direcciones de memoria, las interrupciones (IRQ's) y los canales DMA, manejados por los dos dispositivos.

La opción de Custom Installation presenta las direcciones de memoria, IRQ's y los canales DMA utilizados y disponibles marcándolos con rojo y verde respectivamente. Uno debe manualmente seleccionar las opciones posibles para lograr el funcionamiento de la tarjeta y cada vez que se elige una configuración se instalan los archivos de RealMagic al disco duro y se actualizan los archivos FMPDRV.INI, CONFIG.SYS, AUTOEXEC.BAT, SYSTEM.INI y WIN.INI.

Para verificar el correcto funcionamiento de la tarjeta, existe una utilidad TEST en el directorio C:\REALMAGIC que nos permite seleccionar un archivo .MPG y reproducirlo. Cuando se logra reproducir un video desde DOS, podemos entrar a Windows, donde encontraremos un grupo de programas llamado RealMagic que contiene ciertas utilidades para controlar el volumen de reproducción o ajustar el borde de la pantalla donde se reproducirán los videos. Para reproducir videos en ambiente Windows se utiliza el Transmisor de Medios (Media Player), el cual es una pequeña aplicación que nos permite seleccionar el video a reproducir y cuenta con botones de reproducción, pausa, parar, etc. simulando una grabadora.

Para reproducir videos desde Visual Basic se utiliza el control de Multimedia MCI el cual permite la reproducción y grabado de archivos en dispositivos de Interfaz de Control de Medios (Media Control Interface) como el Transmisor de Medios. Este control no viene incluido entre los controles estándar proporcionados en el paquete de Visual Basic, si no que se encuentra en el paquete Microsoft Multimedia Development Kit. La característica principal de este control es que permite utilizar el Transmisor de Medios mediante comandos.

Para la reproducción del video se colocó un objeto imagen que ocupara toda la pantalla y encima de ésta se colocó otra imagen transparente que pudiera detectar cuando el usuario tocaba la pantalla y mediante el evento CLICK, detener la proyección del video e iniciar el flujo del programa. El inicio de la reproducción es activado por un objeto reloj

que al detectar que no ha sido utilizada la terminal por espacio de 60 segundos inicia el evento PLAY del control multimedia.

MONITOR

El monitor fabricado por American Action INC., es del tipo CRT, de 9 pulgadas. Soporta resolución SVGA. La elección de este tamaño de monitor fue para reducir el tamaño de la terminal y para que el usuario al hacer una consulta de información confidencial, pudiera cubrir con su cuerpo el campo visual de la pantalla y evitar que una persona observe las operaciones realizadas.

Sus características son:

Despliegue de colores	Ilimitado
Dot pitch	0.28 mm
Screen	Non - glare etched
Tamaño del despliegue	160 x 120 mm
Resolución	800 x 600, 1024 x 768 interlaced
Frecuencia horizontal	30 - 38 Khz
Frecuencia vertical	47 - 90 Hz
Entrada de video	análoga
Voltaje de alimentación	100 - 240 VAC
Consumo	60 W máximo

PANTALLA SENSIBLE AL TACTO

La pantalla sensible al tacto fabricada por Dynapro Thin Film Products INC., es del tipo "Analog Resistive Touch Screen (ARTS)", el cual provee una interfaz amigable para la interacción hombre máquina, pudiendo ser operado, sólo con la presión de los dedos, o con cualquier elemento cuyo punto de contacto con la pantalla tenga una superficie suave y redondeada. La pantalla sensible al tacto consta de dos capas conductoras transparentes que forman un arreglo de contactos ordenados por filas y columnas, separadas por un polímero en cuyos puntos de intersección presenta una zona de contacto eléctrico la cual se activa bajo presión.

Por las características del polímero usado y sus recubrimientos, el ARTS es resistente a la humedad, a la condensación y al polvo. Debido a su construcción, la posición de sus contactos es fija, lo cual garantiza una permanente disposición física entre la zona de contactos y la presentación gráfica del display.

Los recubrimientos posteriores, junto con dar la protección mecánica, poseen propiedades que permiten reducir el brillo (anti - glare coating).

Sus características principales son:

Durabilidad

> 1 millón de contactos por punto

Recorrido de contactos	0.001" típico
Rebote de contactos	< 10 msec
Transmisibilidad	> 86 % @ 550 nm
Reflectibilidad	< 25 % @ 550 nm
Color	Neutro

La instalación de los drivers de la pantalla sensible al tacto son como los de cualquier mouse tanto en sistema operativo como en Windows, con un diskette proporcionado por el fabricante. Una vez instalados los drivers para la pantalla, se debe correr una utilidad para calibrar las posiciones de la pantalla, la cual consiste en indicarle manualmente cual es la parte superior izquierda y la parte inferior derecha de ésta, para que se pueda hacer un correcto mapeo de las posiciones al seleccionarl

LECTOR DE TARJETA CON BANDA MAGNETICA

El lector de tarjeta con banda magnética marca JARLTECH, es del tipo deslizante, lo que implica que la tarjeta debe ser deslizada por el cabezal del lector. Se eligió este tipo de lector por que en los que se inserta la tarjeta requieren más mecanismos y ocupan mayor espacio.

Sus características son:

Capacidad de decodificación Track simple, track1, track2, track3 y dual track

Velocidad de lectura	7.5 - 8 cm/seg.
Status indicador	Emite dos "beeps" por cada lectura correcta
Interfaz	IBM PC/XT/AT, IBM PS2, Keyboard emulator
Dimensiones	100 L x 42 W x 27 H (mm)
Peso	100 g
Voltaje	+ 5 VDC desde la computadora

El lector de tarjeta magnética es completamente modular, lo que permite una fácil remoción o reemplazo con fines de mantenimiento. Su durabilidad típica es de 300,000 lecturas.

La entrada de información es a través del puerto de teclado y al deslizar la tarjeta, la información se recibe como si se hubiera ingresado el contenido grabado en la banda magnética en el teclado. Es necesario detectar el carácter 13 que indica una nueva línea, ya que de esta manera sabremos que terminó un track de la tarjeta (la tarjeta cuenta con 3 tracks pero únicamente se utilizan 2 para guardar información). Por esto para recibir la información de la tarjeta se debe crear un objeto de caja de texto por cada track que se quiera leer. Para evitar que la información de la banda magnética aparezca en la pantalla se crea una caja de texto con fondo negro, de tamaño lo más reducido posible y en una zona de la pantalla donde exista un objeto negro.

IMPRESORA

La impresora es de tipo matriz de punto, marca Citizen modelo DP-614 DFC, bidireccional. Como elemento adicional incorpora un mecanismo de autocorte de papel, de prepicado (es decir puede cortar el papel solo en ciertas partes para que el usuario sólo tenga que desprenderlo) y un detector de poco papel en el rollo.

Sus características más relevantes son:

Composición del carácter	7 x 7 puntos
Número de caracteres por línea	40 columnas, 360 puntos/línea
Velocidad de impresión	3 líneas por segundo
Tamaño del carácter	1.36 x 2.2 mm
Tipo de papel	Rollo de 76 mm de ancho y 180 mm de diámetro máximo
Interfaz de comunicación	Paralela (8 bits centronic)
Cinta impresora	Cartridge de cinta negra, negra/roja o púrpura

La fuente de poder de la impresora, es diseñada y fabricada por DTS Limitada, y está dedicada a proveer alimentación a la impresora, para lo cual entrega 24 VDC/0.8 [A]

La impresora cuenta con códigos definidos para cambiar el tamaño de letra o hacerla más gruesa lo cual fue necesario para la impresión del título con el nombre del banco. Además, al terminar la impresión, se enviaba el código para cortar el recibo y se imprimían 7 avances de línea para que el recibo pudiera quedar exactamente en la ranura de salida de la impresora.

DISCO DURO

El disco duro es de 3 1/2" de tamaño, con una capacidad de almacenamiento de 512 Mb, tiempo de acceso de 14 ms y la rapidez de transferencia 2.83 MB/s. Como se puede observar la capacidad del disco no es muy grande, pero es suficiente para los requerimientos del sistema y podría ser una opción para la utilización de discos duros de máquinas viejas que ya no tengan uso.

FUENTE DE PODER PRINCIPAL

Esta fuente es del tipo switching de 200 Watts de potencia (Standard de PC), su alimentación es de 220 VAC o 110 VAC y tiene como salidas + 5 VDC/20 [A], - 5 VDC/0.5 [A], + 12 VDC/8 [A] y - 12 VDC/0.5 [A].

Esta fuente alimenta al MotherBoard, al disco duro, al Touch Screen, al controlador de la impresora y al lector de tarjetas magnéticas.

MONTAJE DE LA TERMINAL

La terminal es un equipo diseñado para ser instalado empotrado a una pared, con el objeto de aprovechar al máximo el espacio.

Para el montaje del equipo, se requieren dos pernos de cabeza redonda de 1/4" de diámetro y 1" de largo. La distancia requerida entre los pernos es de 23 cm, y la altura recomendada es de 109 cm.

LECTOR DE CÓDIGO DE BARRAS

El lector de código de barras es de la marca SpectraPhysics modelo SP*ACE y se conecta directamente al puerto serial. Consta de un patrón de reconocimiento de 11 líneas de rayos laser en forma de asterisco (*) generado por un diodo laser visible. El SP*ACE, produce un volumen de reconocimiento de 70 pulgadas cúbicas y puede ser utilizado para leer códigos de barras de hasta una distancia de 8 pulgadas (20.3 cm).

Sus características más relevantes son:

Dimensiones:	15.2cm H x 10.9cm W x 7.9cm H
Peso:	552 g.
Voltaje (nominal):	+12 V
Corriente (nominal):	330 mA
Poder (nominal):	4 W

Fuente de luz:

diodo laser visible de 670 nm

Para la lectura del código de barras se tuvo que utilizar el objeto de comunicaciones, el cual nos permite establecer la interfaz entre el puerto serial y el lector. Para esto es necesario indicar los parámetros de comunicación como son velocidad, paridad, longitud de palabra y bit de parada. El lector de código de barras también debe ser programado, y para esto el proveedor proporciona un manual con varios códigos de barras para establecer la configuración necesaria.

Una vez establecidos los parámetros de comunicación también se debe indicar el formato de código de barras a utilizar en el lector, ya que existen estándares internacionales para la definición de códigos de barras, además de que hay que indicar la cantidad de barras o dígitos a leer, que puede ser de 24 para la compañía de luz ó 21 para el recibo telefónico.

Cuando los parámetros de comunicación ya fueron establecidos en ambos equipos, se deben enviar ciertos códigos definidos para recibir la información en la PC. Si se recibió la señal que indica que una lectura fue realizada, es necesario verificar que el código no contenga caracteres como Avance de Línea o Enter.

Capítulo 5

Etapa de Pruebas

5.1 ASEGURAMIENTO DE LA CALIDAD

El aseguramiento de la calidad se refiere al proceso de revisión de los productos de software y a la documentación relacionada para que sean completos, correctos, confiables y fácil de mantener. Este proceso incluye, así mismo, el aseguramiento del cumplimiento de los sistemas de las especificaciones y requerimientos que se proyectaron para su uso y desempeño.

5.2 NIVELES DE SEGURIDAD

Los analistas utilizan cuatro niveles para asegurar la calidad: prueba, verificación, validación y certificación.

5.2.1 PRUEBA

La prueba de sistemas bancarios es un proceso caro pero crítico, que puede utilizar hasta 35% del presupuesto del desarrollo de los programas. En general, la prueba de programa se lleva a cabo para probar que no existen errores en un programa. Sin embargo,

como se indicó anteriormente, esto es imposible, dado que los analistas no pueden probar que el software esté libre y limpio de errores; por lo tanto, el enfoque más útil y práctico es entender que la prueba es el proceso de ejecutar un programa con la intención específica de encontrar errores; es decir, hacer que el programa falle. La persona que prueba, que puede ser analista, programador o un especialista entrenado en la prueba del software, en realidad trata de que el programa falle. Una prueba exitosa, por lo tanto, es aquella en que se encuentra un error.

5.2.2 VERIFICACIÓN Y VALIDACIÓN

Al igual que la prueba, la verificación también se utiliza con objeto de encontrar errores. Se desarrolla al ejecutar un programa en un ambiente simulado. La validación se refiere al proceso de utilizar el software en un ambiente real para encontrar errores.

La retroalimentación de la fase de validación produce cambios en el software para manejar los errores y fallas que se descubren.

El sistema está en "vivo" en todo el sentido de la palabra excepto que los usuarios saben que están utilizando un sistema que puede fallar. Las transacciones que se introducen y las personas que los utilizan son reales.

La validación puede continuar por varios meses. Durante el curso de la validación del sistema, la falla puede ocurrir y el software será modificado. El uso continuo puede producir fallas adicionales y la necesidad de más cambios.

La etapa de pruebas del sistema se dividió en 5 fases que son las siguientes:

- 1.-Prueba unitaria de programas
- 2.-Pruebas integrales por módulo
- 3.-Pruebas integrales del sistema
- 4.-Prueba de desempeño en tiempo.
- 5.-Prueba de recuperación

1. Prueba unitaria de programas.

La prueba unitaria se enfoca primero a los programas, independientes uno de otro, para localizar errores, lo que permite a quien está haciendo la prueba detectar errores de codificación y lógica que están contenidos dentro de cada uno de los programas. Al inicio se evitaron aquellos que resultan de la interacción entre los diferentes módulos.

Por ejemplo, los primeros programas que se probaron fueron reportadores, es decir todos aquellos programas que mostraban una salida de un pequeño reporte en papel, esto es por que la información que obtiene este programa tan sólo es formateada e impresa, no tiene que ver con la conexión con otros módulos. Después se probaron las rutinas de

servicio, después todos aquellos programas que mostraban datos en las pantallas y al final los que interactuaban con otros programas y los que realizaban procesos internos del sistema.

Los casos de pruebas necesarios para las pruebas unitarias ejercitaron cada condición y opción, con esto los programas eran exhaustivamente probados, siguiendo el flujo de las posibles opciones que se podrían presentar en un programa. También se probó con los valores límites esperados, incluyendo tanto valores válidos como inválidos.

Si el programa estaba diseñado para desempeñar iteraciones, con procesos específicos contenidos en cada ciclo se probó con las condiciones límite: cero iteraciones, una iteración a través del ciclo y el número máximo de iteraciones.

Alguno de los problemas con los que nos enfrentamos en esta etapa, fue en probar aquellos programas que recibían información del Host, ya que dependíamos de que los servidores estuvieran activos para poder hacerles la petición desde la PC, y si no contábamos con conexión, era necesario indicar el problema al usuario y guardar la información en la bitácora para posteriores aclaraciones.

2.-Pruebas modulares.

En esta etapa se prueban el conjunto de programas que conforma un módulo, por ejemplo, pago de luz, que está formado de varios programas que tienen una tarea específica, pero que interactúan dependiendo del buen funcionamiento de los otros programas.

Se realizó una matriz de pruebas para probar el ciclo completo de cada una de las operaciones relacionadas con cada módulo. Lo anterior permitió ver que errores tenían los programas al momento de tener una conexión con los otros programas del mismo módulo. En esta etapa se encuentran más errores que no fueron detectados en la etapa anterior ya que al probar el ciclo completo del módulo se pueden ver opciones que no se habían contemplado antes.

La preocupación principal en esta etapa fue verificar la compatibilidad de los programas individuales.

En esta etapa se acercaron los datos lo más posible a la realidad ya que de alguna manera nuestro sistema se iba acercando cada vez más a la instalación en producción.

3.-Pruebas integrales del sistema.

En esta etapa se empezó a ver como funcionaría el sistema en producción, ya que todos los módulos están listos y probados individualmente, lo que lleva a pensar que el

sistema está listo, sin embargo este tiempo fue de lo más crítico ya que se empezó a notar que ciertos módulos tenían errores con la interacción de otros módulos, es decir, se presentaron errores que sólo se podían haber presentado cuando todos los módulos se pusieran en marcha, como es el caso de las consultas, en ocasiones el sistema confundía las cuentas asociadas a un cliente cuando este había realizado una consulta anterior en otro tipo de cuenta. Además ya que el usuario se empezó a relacionar más con el sistema, se dio cuenta de que necesitaba algunos detalles más, con esto, se pidió al equipo de desarrolladores agregar los detalles que se le habían pasado al usuario.

La prueba consistió en ejecutar el sistema como si se tratara de una situación real, teniendo todos los módulos activos para que respondieran en el momento adecuado. Para esto, previamente se realizó una matriz de pruebas que contemplaba el universo de posibilidades que podía surgir en la vida real. La matriz fue muy importante dado que sirvió para darnos cuenta que aunque los programas funcionaban bien individualmente y por módulos, la interacción entre ellos mostró que faltaba tomar en cuenta algunas condiciones que en otro momento no se presentaron. Aunque la matriz tenía un buen acercamiento a la realidad, también se presentó que faltaban casos que no se presentaban en dicha matriz y que sólo la experiencia del usuario nos los podría decir. Dada la experiencia de cada una de las personas involucradas en el equipo, los problemas se resolvieron rápidamente conforme se fueron presentando.

Algunas de las partes más críticas fue la comunicación con el sistema central, ya que al principio no se podía hacer conexión con éste debido a que aún le faltaban detalles de arreglar al disparador de transacciones del mismo.

4.- Prueba de desempeño en tiempo

Esta prueba fue realizada tomando en cuenta los días y horas pico, lo cual arrojó en promedio un tiempo de respuesta de 10 seg., siendo este el tiempo promedio máximo de atención, lo cual nos marca un parámetro para considerar el tiempo en el cual una transacción se puede marcar como no concluida. En base a esto, el tiempo máximo de respuesta del host se estableció en 30 segundos y se colocó un contador para que si en este tiempo no se recibe respuesta, se cancele la transacción y se guarda un registro en la bitácora.

5.- Prueba de recuperación.

Para esta prueba se siguió todo un planteamiento de posibilidades, ya que había bastantes variantes que podían alterar el buen funcionamiento de las conexiones entre las terminales y la computadora central, el rompimiento de un enlace a mitad de una transmisión, la falta de confirmación para una operación dudosa. Ya que el hecho de perder la comunicación entre la PC y el Host no es una posibilidad remota, teníamos que estar seguros que el sistema estaba preparado para una interrupción en la comunicación. Había bastantes casos críticos con respecto a este error, pero uno de los más significativos, es

aquel en el cual el cliente está operando una transacción y el sistema ya no le responde satisfactoriamente ya que se cayó el enlace; sólo mostrándole en la pantalla que su operación no se efectuó y que si desea puede comunicarse con los ejecutivos que le darán más información acerca de esa transacción, sería posible mantener la confianza del usuario en la terminal de información. En este caso se contempló el hecho de que la PC hiciera 5 intentos por restablecer la comunicación, antes de abortar el requerimiento, si después de 5 intentos la comunicación no se restablece, la PC deja de intentar y marca la operación como inconclusa para que si el cliente efectúa una reclamación ésta pueda ser manejada por el ejecutivo y darle información sobre el estado en que se quedó la transacción.

Capítulo 6

Implantación

6.1 MÉTODOS DE CONVERSIÓN

La conversión es el proceso del cambio del sistema antiguo al nuevo. Existen cuatro métodos para manejar la conversión de sistemas. Cada método debe considerarse de acuerdo con las oportunidades que ofrece y los problemas que pudiera originar. Sin embargo, en algunas circunstancias obligan a que un método se prefiera respecto a otro, aunque otros podrían proporcionar más beneficios. En general, la conversión se debe realizar tan rápido como sea posible; los períodos largos de conversión aumentan la posible frustración y dificultad de la tarea para todas las personas que participan, incluyendo tanto a los analistas como a los usuarios.

6.2 SISTEMAS PARALELOS

El método más seguro de conversión de un sistema viejo a uno nuevo radica en operar ambos sistemas en forma paralela, lo que significa que los usuarios continúan operando el viejo sistema de la manera acostumbrada, pero empiezan también a emplear el nuevo sistema. Este método constituye el enfoque de conversión más seguro, dado que garantiza que en caso de que aparezcan problemas durante el empleo del nuevo sistema,

como errores en el procesamiento o incapacidad para manejar determinados tipos de transacciones, la empresa aún podrá valerse del viejo sistema sin pérdidas de tiempo, ingresos o servicios.

Las desventajas del enfoque de sistemas paralelos son significativas. En primer lugar, los costos del sistema se duplican, ya que existen ahora dos conjuntos de costos de sistemas. En algunos casos, es necesario contratar personal eventual que ayude a la operación de ambos sistemas en paralelo. En segundo lugar, el hecho de que los usuarios saben que pueden apoyarse en los viejos sistemas puede convertirse en una desventaja si existe resistencia potencial para el cambio o si los usuarios prefieren el viejo sistema; dicho de otra manera, el nuevo sistema no será ensayado de manera correcta.

Como resumen, el método paralelo de conversión de sistemas permite la puesta en marcha más segura, en caso de que haya dificultades, pero no se pueden ignorar los costos y riesgos de no ensayar adecuadamente el nuevo sistema.

6.3 CAMBIO DIRÉCTO

El método de cambio directo convierte el sistema viejo al nuevo de manera repentina, en ocasiones en el lapso de un fin de semana o incluso de un día para otro. El sistema viejo se emplea hasta el día de conversión planeado y entonces se reemplaza por el sistema nuevo. No existen actividades paralelas.

Si el analista debe llevar a cabo el cambio y desea garantizar que el nuevo sistema reemplace completamente al viejo, de manera que los usuarios no confíen en los métodos anteriores, el método directo logrará este objetivo. Psicológicamente obliga a todos los usuarios a lograr que funcione el sistema nuevo; no tienen alternativa.

La ventaja de no tener un sistema de apoyo se puede convertir en una desventaja si surgen problemas serios con el sistema nuevo. En algunos casos las empresas pueden incluso, detener las operaciones cuando surgen problemas y éstos se pueden corregir.

El cambio directo requiere de una planificación cuidadosa por adelantado. Las sesiones de capacitación se deben programar y mantener. La instalación de todo el equipo debe ser puntual con fechas precisas dentro del programa, para corregir las dificultades que puedan ocurrir. Cualquier preparación en el lugar debe ser completa antes de que se lleve a cabo la conversión. Las conversiones directas son muy comunes, especialmente con sistemas comprados o alquilados.

6.4 ENFOQUE PILOTO

Cuando los sistemas nuevos implican también técnicas nuevas o cambios drásticos en el desempeño de la empresa se prefiere el enfoque piloto con frecuencia. En este método una versión práctica del sistema se pone en marcha en una parte de la empresa; por ejemplo, una sola área o departamento de trabajo. Los usuarios de esta área saben por lo general que

están llevando a cabo una prueba piloto de un nuevo sistema y que pueden practicarse cambios para mejorarlo.

Cuando se calcula que el sistema está completo, se instala en toda la compañía, ya sea de una sola vez (cambio directo) o en forma gradual (por etapas).

Este enfoque tiene la ventaja de proporcionar un sólido terreno antes de la puesta en marcha total. No obstante, si la puesta en marcha no se realiza de manera adecuada, los usuarios pueden tener la impresión de que el sistema sigue teniendo problemas y que no puede confiarse en éste. Por ejemplo, podría considerarse que las dificultades que tuvieron durante dos o tres semanas de hecho no desaparecen sólo por que los analistas juzgan que han resuelto las dificultades.

6.5 MÉTODO POR ETAPAS

El método por etapas se emplea cuando no es posible instalar un sistema nuevo en toda la compañía de manera simultánea. La conversión de archivos, la capacitación de personal o la llegada de equipo pueden forzar a la puesta en marcha gradual que se realiza en semanas o meses. Algunos usuarios obtendrán ventajas del sistema nuevo antes que otros; por ejemplo, un sistema médico que trata de llevar a cabo una interconexión de 10 ó 15 clínicas diferentes en un hospital, puede distribuir el proceso de cambio por etapas con un año de duración.

El trabajo que se requiere para la conversión de registros en papel sobre la información de pacientes y de los registros de seguros o archivos almacenados en discos magnéticos, requiere de dos o tres semanas para cada clínica. Una semana de capacitación para los usuarios se requiere también para cada clínica, por lo tanto, los analistas pueden dividir en etapas este sistema en una clínica cada vez, dando tres o cuatro semanas para cada conversión. Si el sistema trabaja bien, los primeros usuarios comunicarán su entusiasmo a otros que esperan la puesta en marcha. De hecho, el entusiasmo puede llegar a niveles tan altos, que cuando un grupo de usuarios finalmente reciba el sistema se presenta una desilusión. En el ejemplo de las clínicas el grupo médico puede exagerar los ahorros del tiempo logrados por no tener que buscar en los registros médicos o preparar de manera manual reclamaciones de seguros, lo que realiza el nuevo sistema. Posteriormente, cuando ocurre la conversión, el personal descubre que el sistema aunque trabaja de manera adecuada, no efectúa el procesamiento en manera instantánea. La decepción es comprensible. Por otra parte, si existen problemas al inicio de la puesta en marcha por etapas, se harán comentarios sobre las dificultades. Así, los usuarios pueden esperar problemas cuando se realice la conversión y reaccionar de manera negativa en los errores más pequeños, incluso los que ellos mismos cometen. Cuando en los sistemas se utilicen los métodos por etapas, deben trabajar adecuadamente en la primera conversión y en las que siguen.

6.6 PLAN DE CONVERSIÓN

El plan de conversión incluye una descripción de todas las actividades que deben llevarse a cabo para poner en práctica el sistema nuevo y ponerlo en operación; asimismo, identifica a las personas responsables de cada actividad e incluye una distribución de horarios que estipula cuando ocurrirá cada actividad. Durante las etapas anteriores a la puesta en marcha, cuando la conversión está en la etapa de planeación, los analistas deben preparar una lista de todas las tareas, incluyendo las siguientes:

1. Lista de todos los archivos que se usarán en la conversión
2. Identificación de todos los datos que se requieran para la construcción de archivos nuevos durante la conversión.
3. Lista de todos los documentos y procedimientos nuevos que se usarán durante la conversión.
4. Identificación de todos los controles que se utilizarán durante la conversión.
Establecimiento de procedimientos para la verificación cruzada del sistema nuevo y el antiguo. Determinar la forma en que los miembros del personal puedan saber si algo no se ha completado de manera adecuada.
5. Determinación de responsabilidades para cada actividad.
6. Verificación de programas de conversión.

El plan de conversión debe anticipar los posibles problemas y métodos para enfrentarlos. Entre los problemas más frecuentes se encuentran: documentos faltantes,

formatos con datos cruzados entre los archivos actuales y los nuevos, errores en la traducción de datos, datos faltantes o archivos perdidos y circunstancias que no se previeron durante el desarrollo de sistemas. El gerente del proceso de conversión debe estar alerta en cuanto a la omisión de pasos. Una lista de verificación evitará que se omitan pasos. Las inasistencias del personal también deben contemplarse y es necesario elaborar planes específicos para esa circunstancia.

La determinación de tiempos para la conversión es difícil, ya que existen muchos que varían desde la instalación del equipo hasta la requisición de formas y suministros. Algunas de las actividades de la conversión dan principio, de hecho, mientras el sistema se encuentra en desarrollo.

6.7 ANÁLISIS DE IMPACTO

En el Sistema de Manejo de Operaciones Bancarias se usó una conversión mediante un método piloto. Ya que se trataba de un sistema nuevo y no venía a sustituir ningún otro, el método piloto es el más apropiado.

Las terminales de información para manejo de operaciones bancarias se colocaron en dos sucursales en el área metropolitana. Estas terminales fueron monitoreadas durante el transcurso de su desempeño ante el público.

El día que se puso en marcha el sistema de manejo de operaciones bancarias todos los archivos que se necesitaban estaban creados, todas las conexiones de los sistemas que interactuaban con él, estaban en línea, ya que si cualquiera de estas partes que conforman el sistema no estaba listo, podría causar una falta del servicio, esto implicaba que el cliente no podría tener acceso a su información hasta que todo estuviera completo para funcionar.

Antes que el sistema se mostrara al cliente, se hicieron algunas pruebas en las dos sucursales donde se implementó el piloto, ya que si se detectaba cualquier falla esto podría retrasar la puesta en marcha hasta que los problemas quedaran solucionados.

El sistema no debía por ningún motivo fallar en esta etapa, pues el cliente esperaba que sus transacciones se realizaran normalmente. Era muy importante que pudiera hacer todo sin ningún problema para que el sistema se ganara la confianza del cliente y no su desconfianza.

El sistema como se mencionó anteriormente tiene la cualidad de mostrar en la pantalla una serie de menús que le permiten al cliente navegar entre las diferentes opciones de servicios que se le ofrecen, el cliente podrá realizar sus operaciones sin la intervención de un empleado bancario. El sistema debía de ser capaz de captar su atención y ganar su confianza para que realizara alguna operación. Este momento era muy importante ya que se vería si el sistema podía ganar la aceptación del público en general que llega a la sucursal. También la respuesta ante este plan piloto nos dictará la rapidez con que el servicio en otras

sucursales será instalado, si la aceptación es generalizada y la respuesta del público es favorable, la instalación del servicio en otras sucursales será más rápido.

Durante la etapa de pruebas en las dos sucursales, se comenzó a trabajar con las terminales teniendo algunos problemas de comunicación con el servidor central, los cuales se solucionaron en 3 días. Otro problema fue el de ver que la terminal tuviera los suficientes privilegios para conectarse al host y efectuar transacciones en el sistema de producción. Los demás problemas fueron pequeños errores en el desarrollo del sistema, que sólo fueron detectados al realizar varias transacciones en producción y no en un ambiente de pruebas.

Capítulo 7

Conclusiones

El desarrollo tecnológico en los últimos años nos ha permitido un manejo formidable de la información, por lo cual, es ya muy común encontrar la computadora en cualquier sitio, resolviendo problemas en fracción de segundos. Hay algunos lugares donde todavía la incursión de la computación es vaga aún, sin embargo, hay otros que dependen cien por ciento del uso de una computadora, un ejemplo muy claro es el caso de un banco, que sin computadoras en este instante que vivimos no podría operar. Este desarrollo tecnológico nos va mostrando una gama enorme de posibilidades que pueden ayudarnos a manejar mejor la información. El sistema propuesto usa la última tecnología, un sistema de información que es atractivo al cliente, desde donde puede realizar operaciones de transferencia de fondos, pago de servicios, etc.

El banco se beneficia con este sistema ya que puede tener menos empleados prestando un servicio de información sobre servicios bancarios, además que tendrá menos clientes en las ventanillas realizando operaciones que son más costosas desde este medio.

A futuro se tienen grandes expectativas para este tipo de proyectos, pues en vista que la demanda que se puede obtener por este tipo de servicios es mucha, la posibilidad de

hacer más módulos para el sistema no es remota. Como se mencionó anteriormente el sistema va ir creciendo dependiendo de la demanda que éste genere.

Debido también a la facilidad que van mostrando los nuevos lenguajes, el desarrollo se hace menos pesado y más fácil para los programadores. Prácticamente el usuario puede ver con un prototipo hecho en pocas semanas como va a quedar su sistema, lo cual ahorra muchos malos entendidos en etapas posteriores del desarrollo del mismo.

La competencia entre empresas se vuelve cada vez más fuerte y nadie está exento de perder el mercado que ataca, por esto es importante estar presente y estar a la vanguardia en la tecnología del manejo de la información.

Este sistema de información bancaria es un recurso que nos puede ayudar para múltiples servicios, no tan sólo para las actividades que se han mencionado en el banco, si no como venta de boletos de espectáculos, aviones, etc., tan sólo cambiando las pantallas de presentación y los protocolos de comunicación con el host.

El sistema de información bancaria tiene la ventaja de siempre estar disponible, lo que no ocurre con una persona que atiende en ventanilla.

La metodología que se escogió fue bastante buena, ya que nos llevó a la realización del proyecto en el tiempo que se tenía programado, sin embargo hubo algunos tiempos que se podrían mejorar, como el que se dedicó a realizar la conexión con el host, el cual se llevó

más tiempo del que se había calculado ya que teníamos problemas con el sistema de comunicación red hacia el host.

Con el sistema trabajando como piloto en las sucursales, nos permitió darnos cuenta de que había algunos problemas sin detectar en la etapa de pruebas, gracias a que se presentaron en este momento el impacto en las operaciones es mínimo debido a que el cambio en producción se puede hacer sólo afectando a dos sucursales en operación

Al principio de la operación del piloto se recibieron llamadas de las sucursales, reportando algunas fallas del sistema, sin embargo, con el paso del tiempo, fueron disminuyendo ya que se arreglaron rápidamente.

Uno de los objetivos del proyecto se cumplió casi un mes después de empezada la operación cuando el sistema estuvo funcionando correctamente sin ningún error en días y horas pico. Así mismo otro objetivo se cumplió cuando el número de clientes que se acercaba a la terminal fue aumentando, así como la complejidad de las operaciones realizadas, ya que al principio sólo se realizaban consultas de información y saldos, pero con el paso del tiempo la gente fue adquiriendo experiencia y confianza de tal manera que fue más común ver operaciones de transferencias o pago de servicios, que para el cliente representan más riesgo por que implican traslado de su dinero y podría presentarse el hecho de que se perdiera su transacción. Afortunadamente nunca se presentó un caso de un cliente que una vez que hubiera tenido algún problema con la terminal, éste no hubiera sido aclarado por el personal de la sucursal.

Por lo que se refiere a las Operaciones Financieras, se presentó el caso de que no fueron realizadas por el cliente con mucha frecuencia por lo que se llegó a la conclusión de que el universo de usuarios que realizan éste tipo de operaciones no es tan grande y hay la posibilidad de que en una segunda etapa del proyecto se elimine o substituya esta opción.

En resumen, se cumplieron las expectativas de este trabajo de tesis, que eran el proporcionar al banco un sistema de atención de clientes que fuera fácil de utilizar, a bajo costo, atractivo al cliente y que incorporara elementos de tecnología de punta. La terminal de información cumplió con los requisitos del banco y en la actualidad hay negociaciones para instalarla en las sucursales más importantes de la zona metropolitana.

A raíz de la realización de este trabajo nos hemos dado cuenta del potencial que podrían tener estas terminales realizando diferentes operaciones, como se mencionó anteriormente, una de estas terminales podría dar información acerca de las llegadas y salidas de vuelos que se realizan en un aeropuerto y no tan sólo eso, si no incluso poder vender boletos que se pudieran pagar haciendo a un cargo a una tarjeta de crédito de algún banco. Con este servicio alguna empresa aérea podría poner terminales de atención al público en diferentes lugares, como centros comerciales, turísticos y lugares públicos en general e incluso dentro del mismo aeropuerto, lo cual expandiría sus medios de venta sin necesidad de personal que atienda al cliente. Otro lugar donde podría dar un servicio esta terminal, sería en las tiendas que manejan tarjetas de crédito (SEARS, Suburbia, etc.), ya que el cliente podría pagar su tarjeta de crédito de la tienda, haciendo un cargo a otra tarjeta

de un banco y recibir información de descuentos, promociones y localización de departamentos.

La terminal de información abre al mundo como otra opción al manejo de la información electrónica, las personas poco a poco se están familiarizando con las computadoras, pues es una realidad de nuestro tiempo que ya no se pueden realizar muchos trabajos sin la presencia de ésta, la terminal de información podrá abrir este camino de la confianza en realizar transacciones delicadas como de abonos o cargos a cuentas, trasposos, pago de teléfono, luz, etc. Gracias a la computadora la humanidad ha avanzado más en esta mitad de siglo, que lo que lo había hecho desde el comienzo del primer hombre en la tierra, aún nos falta mucho por recorrer y así como diría el futurólogo Alvin Toffler "en esta carrera sin límites hacia el futuro".

Bibliografía

Classic/PCI LPX Board & Low Profile Desktop Technical Product Summary

Intel Corporation, USA, Version 1.1 May, 1994

Cornell, Gary

Manual de Visual Basic 3 para Windows

McGraw Hill, México, Primera Edición, 1993.

Ehterlink III Parallel Tasking 16-Bit ISA Adapter User Guide

3com Corporation, Santa Clara, CA, USA, October 1995.

Manual de Usuario. Terminal de Autoconsulta Multipropósito INSTANDATA

Desarrollo de Tecnologías y Sistemas LTDA, Santiago, Chile, Revisión 1.0, Abril 1995.

Metodología para la Integración de Sistemas

Integradora de Servicios Operativos, México, v 1.0, 1995

Microsoft MS-DOS y Microsoft Windows para Trabajo en Grupo. Manual de Usuario

Microsoft Corporation, USA, 1994

Programming in Microsoft Visual Basic 3.0. Workbook

Microsoft Corporation, USA, 1994.

Rangel, Raymundo Hugo

Apuntes de programación estructurada

Facultad de Ingeniería UNAM, Primera Edición, 1985.

Real Magic Lite. MPEG Interactive Playback Card. User's Guide

Sigma Designs Inc. Fremont, CA, USA, Rev. A1, December 1994.

Schildt, Herbert

Aplicación Turbo C++

Osborne/MacGraw-Hill, España, Primera edición, 1991.

Senn, James

Análisis y diseño de sistemas de información

MacGraw-Hill, Chalco, Edo Mexico, Segunda edición, 1990.

SP*ACE(tm) Multi-Position Scanner

<http://www.pscnet.com>