



**UNIVERSIDAD NACIONAL AUTÓNOMA DE  
MÉXICO**

**FACULTAD DE INGENIERÍA**

**“SISTEMA DE ATENCIÓN A CLIENTES PARA CONOCER EL  
POSICIONAMIENTO DE PRODUCTOS EN EL MERCADO”**

**TESIS**

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

**PRESENTAN:**

**CABRERA MARTÍNEZ PABLO  
DOMÍNGUEZ ARGUETA MARÍA TERESA  
GARCÍA GARCÍA SERGIO**

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



**CIUDAD UNIVERSITARIA**

**OCTUBRE, 2007**



Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**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.

---

## ***Dedicatoria***

*A mi querida Universidad que fue como una segunda casa, donde conocimos a amigos que nunca olvidaremos.*

*A la persona que algún día me dedico estas frases.*

*El hombre que lucha un día es bueno.*

*El hombre que lucha un año es mejor.*

*Pero el hombre que lucha toda la vida, ese hombre es imprescindible.*

*Dedico mi tesis a las personas que me apoyaron en todo momento de mi vida.*

*A mis padres que siempre confiaron en mí y aunque no podían darme lo que yo quería, pero me dieron lo más importante para mí, mi Estudio*

*A mis hijos que aunque son muy pequeños son los que me dan la fuerza para seguir adelante y trabajar con más ahínco para darles un mejor futuro.*

*A mis hermanos Adela, Juan, Lourdes, Leticia, Ana y Carlos que me apoyaron y se sienten orgullosos de mí.*

*A mis cuñados que son parte de esta familia y me brindaron una palabra de aliento.*

*A mis compañeros y amigos que aunque no están conmigo, Daniel Córdoba Cervantes y Esperanza siempre los llevare en mi corazón.*

*A todos mis sobrinos que los quiero mucho y espero que sigan estudiando.*

*A todos los compañeros de trabajo que me motivaron para salir adelante y terminar mi carrera.*

---

---

*A la persona que más dedico este trabajo es a ti gran esposa y compañera, amante y amiga que siempre estas a mi lado, que soportaste todos mis desvelos, que estuviste conmigo en las buenas y en las malas. Y que siempre te admirare por esa fortaleza que tienes y me trasmitiste*

## ***Agradecimientos***

*A mis padres que me apoyaron en todas las decisiones que tome*

*A la vida que me permitió vivir en esta época tan maravillosa.*

*Al ingeniero Juan Carlos Roa Beiza por guiarnos en este ultimo paso de nuestra carrera.*

*A mis compañeros de tesis Magdalia, Teresa, Pablo y Francisco que me dieron toda su confianza y su amistad.*

***Sergio García García***

---

---

*A mi familia  
Por creer siempre en mí y apoyarme incondicionalmente*

*A mi mamá, por todo el cariño que me das y por ser un ejemplo para salir siempre adelante sin desfallecer.*

*A mis hermanos por que siempre tienen una palabra de aliento y por todo el amor que me brindan.*

*A Pedro por estar conmigo a pesar de los momentos difíciles por los que hemos pasado.*

*A Pablo, Sergio, Ernesto y Magda por su compromiso en la realización de este trabajo, por su amistad y los momentos gratos que pasamos. Creo que formamos un buen equipo!*

*Al PAT y al Ing. Roa Beiza que nos apoyaron para el desarrollo de nuestra tesis, que después de mucho tiempo, por fin se hizo realidad!*

***María Teresa Domínguez Argueta***

---

---

*Dios gracias por apoyarme, cuidar de mí y por todo lo bueno de esta vida.*

*A mi familia por creer y confiar en mí, esto es apenas una pequeña parte del fruto de sus esfuerzos.*

*Mamá muchas gracias tu cariño, dedicación y esfuerzo que realizaste para que este sueño se concretara.*

*Papá gracias por tu apoyo y a mi hermanita por alegrarme siempre la vida.*

*Georgina gracias por tu amistad y cariño que en las buenas y en las malas nos hemos acompañado siempre.*

*A mi gran amigo Lucio que estuvo en mis buenos y malos momentos de la carrera y por toda su amistad que no ha decaído desde que éramos niños.*

*A Teresa, Magdalia, Sergio y Ernesto por aguantarme y brindarme su amistad en este corto tiempo de conocernos.*

*Y a todos, es un gran honor ser egresado de la Universidad Nacional Autónoma de México*

***Pablo Cabrera Martínez***

---



---

## ÍNDICE

Página

### **CAPÍTULO 1. SITUACIÓN ACTUAL**

1.1 Introducción .....	3
1.2 Factores para la utilización de un Call Center.....	7
1.3 Proceso actual del levantamiento de información.....	11
1.4 Proceso actual de administración de la información.....	14
1.5 Costos de operación .....	16

### **CAPÍTULO 2. TEORÍA BÁSICA**

2.1 Metodologías de bases de datos relacionales.....	23
2.2 Bases de datos y herramientas para desarrollo en web.....	36
2.3 Características, ventajas y desventajas de Active Server Page (ASP)	48
2.4 Características, ventajas y desventajas de SQL Server 2000.....	57
2.5 Características, ventajas y desventajas de la comunicación vía Web	66

### **CAPÍTULO 3. ANÁLISIS**

3.1 Problemática actual.....	85
3.2 Requerimientos generales y específicos.....	90
3.3 Análisis de la información recabada.....	93
3.4 Identificación del problema.....	97
3.5 Justificación de la solución propuesta.....	100



---

## **CAPÍTULO 4. DISEÑO E IMPLEMENTACIÓN**

4.1 Diseño del sistema .....	113
4.2 Desarrollo de la interfaz del usuario (front-end) .....	140
4.3 Desarrollo de los procesos en la base de datos (back-end) .....	148
4.4 Generación de reportes .....	168
4.5 Integración y pruebas del sistema .....	170
4.6 Optimización de SQL Server (SQL Server Tuning) .....	184
4.7 Liberación del sistema .....	190
<b>CONCLUSIONES .....</b>	<b>199</b>
<b>MANUAL TÉCNICO .....</b>	<b>203</b>
<b>MANUAL DE USUARIO .....</b>	<b>223</b>
<b>GLOSARIO DE TÉRMINOS .....</b>	<b>231</b>
<b>BIBLIOGRAFÍA .....</b>	<b>237</b>





## PRÓLOGO

El presente trabajo tiene como objetivo la creación de un sistema destinado a mejorar e incrementar la eficiencia operativa de un **call center (centro de llamadas)**, aprovechando los recursos humanos con los que cuenta, para posteriormente utilizarlos en otras áreas y como resultado reducir costos en campañas de diferentes empresas que utilizan sus servicios.

El call center carece de un sistema automatizado para el registro y búsqueda de clientes y productos, por lo que el éxito obtenido hasta el momento, se basa en la grandeza de su capital humano, recurso que financieramente ya no puede ser sostenido debido a la alta competitividad que existe en el mercado.

Nuestra tarea, es desarrollar un sistema que ayude a automatizar las campañas del cliente principal del call center y a partir de éste, definir los estándares necesarios para aplicaciones del mismo tipo, esto implicará la reutilización de algunos módulos que se desarrollarán a lo largo de este sistema.

A continuación se presenta un breve bosquejo del desarrollo del presente trabajo: en el primer capítulo, se define que es un call center y cuales son sus características, se explica el proceso actual del levantamiento y administración de la información, y cómo es procesada sin que exista un sistema que ayude a que esta tarea sea confiable. Además se presentan los costos que todo este proceso implica, los cuales deberán reducirse con la implementación del sistema.

En el segundo capítulo se realiza un análisis de las metodologías de bases de datos, se evalúan las diferentes herramientas para desarrollo en web y bases de datos que existen en el mercado y se eligen las más adecuadas para el desarrollo del sistema.



En el tercer capítulo, se identifican los problemas actuales del call center y se proponen soluciones, se analiza como llega la información, cuáles son las fases por las que pasa hasta procesarse y los requerimientos que ésta debe cumplir para obtener los resultados esperados. Una vez recopilada toda la información, se analiza cual es la mejor manera en la que se procesará.

En el cuarto capítulo se realiza el diseño del sistema utilizando la Metodología Yourdon/De Marco, analizando flujos de entrada y salida de la información, diagramas de procesos, diagramas entidad-relación del sistema y la normalización de la base de datos que se utilizará. Se implementa la interfaz gráfica que utilizará el usuario y se explica el uso de las herramientas para la creación de la base de datos.

Una vez desarrollado el sistema se pasa a la etapa de pruebas, tanto de la aplicación web como del servidor de base de datos y se hace la afinación del software. Finalmente, se realiza un contrato de servicio para delimitar las acciones que se llevarán a cabo por parte de ingeniería de sistemas a partir de la liberación del sistema.

# **CAPÍTULO 1**

## **SITUACIÓN ACTUAL**



### 1.1 INTRODUCCIÓN

Actualmente muchas empresas están haciendo uso de los servicios de un **call center (centro de llamadas)** para dar respuesta a las múltiples necesidades que se crean en torno a la atención al cliente. Sin embargo, la realidad nos demuestra que la tecnología está preparada para ofrecer respuestas mucho más satisfactorias que las que actualmente obtienen las empresas.

Si se contara con sistemas de atención al cliente con los que se pudiera automatizar y agilizar el levantamiento y obtención de información, se podría evitar la obsolescencia tecnológica, apoyando la modernización y tecnificación de los mismos con nuevas herramientas para la optimización de la gestión de los recursos humanos, la gestión de la calidad, del conocimiento y de la información para una mejora de la eficiencia/eficacia del centro, y permitir estrategias sin papeles, así como funciones de marketing dinámico y atención de campañas comerciales en tiempo real.

Dichos centros nacieron de la oportunidad de prestar un servicio inmediato al cliente a través del teléfono, al principio era principalmente informativo y tenía un carácter de servicio accesorio a la oferta principal del producto. Sin embargo, se expandió considerablemente, debido principalmente a dos factores:

- Fuerte competencia: que convirtió un servicio de lujo en un canal habitual y necesario de contacto con el cliente.
- Fuerte demanda del cliente particular: que cada vez goza de menos tiempo de ocio y por tanto le da más valor a su tiempo libre.

Pero si hubo y hay un factor clave, es el avance tecnológico, el cual está ayudando a reducir considerablemente el tiempo de respuesta. Esto hace que el

---



número de tipo de productos y servicios ofrecidos y realizados a través del teléfono aumente, llegando a cubrir prácticamente todo el ciclo de relación con el cliente, desde el marketing hasta el servicio postventa, pasando por la venta y todo a través de un solo número de teléfono.

Pero la evolución continúa, y el mercado tan competitivo en el que nos movemos ha enseñado a los usuarios a reclamar servicios de valor agregado y a exigir la forma en cómo quieren relacionarse con la empresa. Esto ha obligado a integrar diversos canales de interacción con la empresa como teléfono, fax, correo electrónico, etc., para proporcionar una solución de centro de atención telefónica y ofrecer a los clientes un único punto de contacto para resolver sus necesidades.

El principal indicador de calidad de dichos centros, es el grado de satisfacción de los clientes y el nivel de solución de incidencias en la primera llamada. Desde el punto de vista del proceso diario, los parámetros a medir son: el número de llamadas realizadas, cuántas han sido retenidas y el número de clientes que han desistido por el tiempo excesivo de espera. En cuanto a la percepción de la calidad de los agentes, los indicadores son: el tiempo útil conectado, las llamadas atendidas y el número de llamadas abandonadas por el agente.

También hay que tener en cuenta que a menudo, estos centros son el único punto de contacto e interacción de un cliente con la compañía, por lo cual la imagen que se formen los consumidores de ellos, será identificada al cien por ciento con la que tengan de la empresa.

Las compañías a nivel mundial han mostrado una tendencia por la utilización de servicios de terceros y en este caso, el uso de los call centres genera ahorros en los costos de entre 30 y 40 por ciento.



En el mundo actual, donde la información se ha convertido en un recurso muy valioso para las empresas, quizás el más importante, es necesario contar con una infraestructura para recabar y procesar información que cuente con los elementos requeridos para proveer información real y oportuna en la toma de decisiones y con esto proporcionar mejores servicios y productos a los clientes.

Con la introducción de internet y de la web en concreto, se han abierto infinidad de posibilidades en cuanto al acceso a la información desde casi cualquier sitio. Esto representa un desafío para los desarrolladores de aplicaciones, ya que los avances en tecnología demandan cada vez aplicaciones más rápidas, ligeras y robustas que permitan utilizar la web.

El desarrollo de aplicaciones web involucra decisiones no triviales de diseño e implementación que inevitablemente influyen en todo el proceso de desarrollo, afectando la división de tareas. Los problemas involucrados como el diseño del modelo del dominio y la construcción de la interfaz del usuario tienen requerimientos que deben ser tratados por separado. El alcance de la aplicación y el tipo de usuario a los que estará dirigida son consideraciones tan importantes como las tecnologías elegidas para realizar la implementación.

Así como las tecnologías pueden limitar la funcionalidad de la aplicación, las decisiones de diseño equivocadas también pueden reducir su capacidad de utilización. El desarrollo de este tipo de aplicaciones debe incluir un proceso general que tenga en cuenta de manera explícita las características particulares de las aplicaciones web.

Una vez que el software ha sido implementado, debe ser probado para descubrir los defectos que puedan existir en la función, en la lógica y en la implementación. Una estrategia de prueba de software integra las técnicas de diseño de casos de

---



prueba en una serie de pasos bien planificados, que dan como resultado una correcta construcción del software.

La prueba del software es un elemento crítico para la garantía de calidad del software y representa una revisión final de las especificaciones, del diseño y de la codificación. Se han propuesto varias estrategias de pruebas de software que tienen las siguientes características generales:

- La prueba comienza en el nivel de módulo y trabaja hacia fuera, hacia la integración de todo el sistema.
- Diferentes técnicas de prueba son apropiadas en diferentes momentos.
- La prueba la lleva a cabo el que desarrolla el software y un grupo de prueba independiente.

La fase de mantenimiento se centra en el cambio que va asociado a la corrección de errores, a las adaptaciones requeridas por la evolución del entorno del software y a las modificaciones debidas a los cambios de los requisitos del cliente dirigidos a reforzar o a ampliar el sistema. Dentro de la fase de mantenimiento se encuentran tres tipos de cambio:

- Mantenimiento correctivo – es el proceso que incluye el diagnóstico y la corrección de uno o más errores.
- Mantenimiento adaptativo – es una actividad que modifica el software para que interaccione con su entorno cambiante.
- Mantenimiento perfectivo – se reciben de los usuarios recomendaciones sobre modificación de funciones ya existentes y mejoras en general.



## 1.2 FACTORES PARA LA UTILIZACIÓN DE UN CALL CENTER

### Definición de un call center

Un centro de llamadas o call center es una plataforma telefónica que tiene la función de facilitar y mejorar la comunicación entre una empresa y sus clientes, optimizando los recursos de esta con el fin de proporcionar mayor valor agregado al cliente y por tanto aumentar su competitividad.

El call center supone una integración de una computadora y un teléfono con el fin de facilitar al cliente información y servicios como lo serían consultas de cuentas bancarias, pagos a proveedores, pedidos, quejas, etc.

Es una estructura de hardware y software que combinando características de comunicación telefónica integrada con la informática del centro de llamadas entre agentes, se orienta a potenciar las labores más importantes de una empresa con la capacidad de asesorar y atender cualquier inquietud de los usuarios además de contestar llamadas.

En resumen podemos definir que es una unidad o departamento en una empresa (o es una empresa especializada) que se dedica al cumplimiento de las funciones de comunicación de está. Las relaciones que pueden establecerse como un medio de comunicación externa en las empresas son:

- Relación con el usuario
- Relación con el cliente
- Funciones de marketing





### **Elementos necesarios de un call center**

Los elementos van a depender de los requerimientos de cada una de las empresas, los más comunes son los siguientes: (ver Figura 1.2.1)

**Central telefónica (PBX, Private Branch Exchange).** Central de conmutación de llamadas telefónicas. Es en esta central donde se van a concentrar todas las llamadas recibidas, el PBX es la parte esencial donde se pueden generar todos los cambios a futuro según las necesidades de la empresa.

**Enlaces troncales.** Son conocidos como E1's, son enlaces de proveedores de telefonía (Telmex, Avantel, Axtel) que dan la capacidad de 30 troncales o líneas por cada E1, las cuales deben ser recibidas y configuradas en el PBX. Las líneas provistas para este tipo de servicios deben ser digitales pero también deben considerarse algunas líneas análogas para servicios especiales.

**Distribución automática de llamadas (ACD, Automatic Call Distribution).** Permite enlazar cada llamada entrante con el agente más apropiado para atenderla, establecer colas de espera, reproducir música o mensajes publicitarios. El software de ACD es vital para la empresa, ya que permite hacer el análisis de tráfico e impacto de respuesta de las diferentes campañas en operación.

**Aparatos telefónicos.** Los aparatos de teléfono para los agentes deben ser compatibles con el PBX y con el tipo de líneas a trabajar, ya sean análogas o digitales, además deben cumplir con características específicas de pantalla y programación de funciones como: not ready, login, logout, conferencia, transferencia de llamadas, así como de tener capacidad para implementar accesorios como diademas para proporcionar comodidad a los agentes.

**Sistema interactivo de respuesta de voz (IVR, Interactive Voice Response System).** Es un conjunto de hardware y software que se encargan de la gestión de

---



Llamadas **inbound (entrantes)** de una organización. Es una unidad que tiene la capacidad de tomar decisiones a partir de ciertos datos que el cliente ingresa y los relaciona con una base de datos.

**Servidor de integración computacional y telefónica (CTI, Computer and Telephony Integration).** Es la capacidad de integración entre el ambiente de telefonía y el software de la interfaz para los operadores.

**Personal humano, operadores o agentes.** La calidad del servicio la brindan los diferentes tipos de agentes debidamente capacitados para su área de campaña, ya que proporcionan respuestas a las peticiones por teléfono de las personas impactadas por la publicidad en diferentes medios de comunicación.

**Zona de trabajo para los agentes.** Cada uno de los puestos de operación donde se ubican los agentes telefónicos para realizar su trabajo de interacción con los clientes debe ser cómoda, iluminada, contar con herramientas de trabajo como PC, aparato telefónico, manuales de sistema, telefonía y guiones. Debe ser una concentración de islas y posiciones distribuidas en un área exclusiva que evite distracciones y facilite la supervisión y el control del personal.

**Servidores de bases de datos.** Son computadoras dedicadas para almacenar la información de los clientes (SQL Server, Oracle, Sybase, etc.).

### **Beneficios de un call center**

- Contribuyen significativamente al incremento de las ventas.
- Permiten un acercamiento más profundo a las necesidades del cliente.
- Establecen un contacto directo e interactivo con el mercado.
- Mejoran el servicio al cliente.



- Amplían la cobertura del mercado.
- Minimizan costos y favorece la rentabilidad.
- Evitan intermediarios.

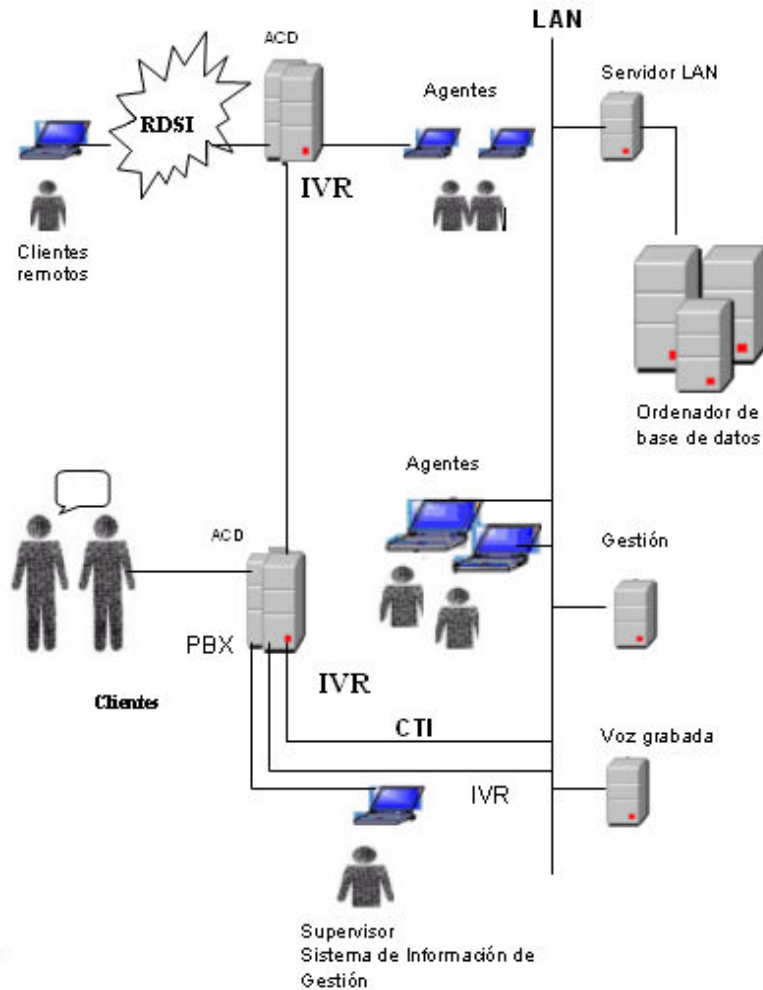


Figura. 1.2.1 Diagrama de un call center



### 1.3 PROCESO ACTUAL DEL LEVANTAMIENTO DE INFORMACIÓN

El call center tiene un total de 72 empleados, entre ejecutivos, supervisores y coordinadores. El tipo de trabajo de cada uno de ellos se clasifica en 3 categorías: por el tipo de servicio que prestan, por la cantidad de horas que trabajan continuamente, y por el número de turnos al día. Dados estos parámetros, la empresa clasifica a su personal en cuatro grandes grupos, los cuales se encuentran enmarcados de la siguiente forma:

- Grupo 1. Son empleados que laboran en horas diurnas, encargados de las llamadas normales, este grupo está conformado por 35 operadores de los cuales 26 tienen contrato por 4 horas, y 9 por 6 horas.
- Grupo 2. Estos empleados se encargan de las supervisiones, este grupo lo conforman 16 operadores, de los cuales 12 tienen contrato por 6 horas y 4 de ellos por 8 horas.
- Grupo 3 y 4: Son los empleados que atienden casos especiales, estos dos grupos se encuentran conformados por 21 operadores, de los cuales 7 de ellos tienen horario de 4 horas y los 14 restantes de 6 horas.

Se destinan recursos para la cobertura de las llamadas telefónicas dependiendo de la importancia de la campaña de que se trate y para determinar esto, se toman en cuenta:

- El análisis de los resultados obtenidos en años anteriores para la misma campaña o en caso de no tener antecedente, de alguna similar.
- El número de operadores disponibles en el momento del lanzamiento de la campaña.



- El pronóstico de demanda planeado por el personal encargado de la realización de la campaña.
- El número de computadoras disponibles en ese momento.

El proceso de levantamiento de información tiene las siguientes características: el cliente realiza una llamada hacia el centro telefónico, un IVR se encarga de filtrarla y por los últimos 4 dígitos del **DNIS (Dialed Number Identification Service)** se le enruta a un operador. A este proceso se le conoce como llamada inbound, donde generalmente son llamadas para registrar productos, solicitar información o bien participar en las diferentes promociones que se ofrecen.

El ejecutivo en turno, debe recabar en hojas de cálculo la información siguiente de la persona que realiza la llamada, sin importar que haya llamado anteriormente:

- Fecha del reporte
  - Nombre
  - Apellido paterno
  - Apellido materno
  - Número telefónico de casa y lada
  - Número telefónico de oficina y lada
  - Número telefónico móvil y lada
  - Sexo
  - Fecha de nacimiento
  - Código postal
  - Calle, número exterior e interior, colonia, delegación o municipio, estado
  - Número de boleto/ticket
  - Código
  - Lote
  - Fecha de caducidad
  - Hora del reporte (verificada por el propio operador)
  - Código adicional (si existe)
-



- Fecha de caducidad adicional
- Hora del reporte adicional (verificada por el propio operador)
- Monto

Primero se intenta buscar al cliente por medio de una función de búsqueda, si no es encontrado, ya sea porque no existe o porque hubo un error al capturar sus datos, se inserta un nuevo registro. Una vez hecho esto, se le pregunta al cliente el motivo de su llamada, teniendo diferentes procedimientos según sea el caso.

Si el cliente presenta una queja o sugerencia, se ingresa toda la información a la hoja de cálculo destinada a observaciones, el ejecutivo debe tratar de obtener la mayor cantidad de información del cliente para tener reportes más completos.

En el caso de suscripciones a promociones del fabricante, el operador solicita al cliente los requisitos mínimos para obtener la suscripción, e ingresa toda esta información en otra hoja de cálculo, algunos de estos datos pueden ser:

- Número de hijos
- Nombre de los mismos
- Edad
- Sexo

En caso de cumplir con estos requisitos, se registran sus datos y se le asigna un número consecutivo, mismo que se le proporciona al cliente con la promesa de enviarle la membresía por correo postal en un período breve.

Si el cliente llama por una promoción, se le pide el nombre de la misma, con esta información el ejecutivo verifica los requisitos y su vigencia; en caso de ser una promoción vencida, se le informa al consumidor la terminación de ésta y se mencionan las que aún están vigentes, o en su caso, se le invita a seguir

---



pendiente de las que habrá en el futuro. Si aún está vigente, se le piden los requisitos mínimos para poder participar en esta promoción; estos datos dependen específicamente de cada promoción y se vacían en otra hoja de cálculo.

El proceso se sigue llevando a cabo durante todo el horario de atención, una vez concluido, cada ejecutivo copia sus hojas de cálculo en una carpeta compartida ubicada en la PC del supervisor en turno. Posteriormente los supervisores envían al **área de servicios de información administrativa (MIS, Management Information Service)** toda la información recabada, en donde se realiza la concentración, depuración y corrección de la misma.

### **1.4 PROCESO ACTUAL DE ADMINISTRACIÓN DE LA INFORMACIÓN**

A lo largo del día, las llamadas recibidas por cada ejecutivo son registradas por el software de distribución automática de llamadas. Esta información es recabada por el área de servicios de información administrativa (MIS), que se encarga de realizar los reportes por día vencido, tanto para el caso de telefonía, como para la manipulación del concentrado de los datos capturados por los operadores. Durante éste proceso se detectan los siguientes casos:

- El cliente llama para solicitar información de la campaña.
- El cliente llama para registrarse y ha proporcionado números telefónicos o domicilios diferentes.
- Se corrigen errores de captura y ortografía.
- Tipificación de llamadas.
- Número de quejas.



- Llamadas y registros por estados.
- Tipo de productos.
- Número de suscripciones.

Asimismo y para fines de control de operación, los datos de telefonía son útiles para optimizar el desempeño de los ejecutivos y por consiguiente un mejor desempeño por campaña, se obtienen métricas para la mercadotecnia del propio call center. Algunas de las métricas que son útiles para el desempeño de una campaña son:

- El **average handling time (AHT - tiempo promedio de llamada)**
- El número de ejecutivos conectados para atención a la campaña.
- El tiempo que permanece en llamada el ejecutivo.
- El número de llamadas abandonadas por ejecutivo y por campaña.
- El nivel de servicio o porcentaje de llamadas atendidas en determinado intervalo de tiempo, el más común es 80/20.

Cabe mencionar, que el reporte de telefonía está estandarizado para todas las campañas que atiende el call center, por lo tanto, para todas se elabora el mismo reporte y con las mismas métricas.

Sin embargo, cada una de las campañas recaba la información acumulada de forma diferente, y es por ello, que existen reportes de clientes, los cuales contienen información que el cliente desea conocer, de acuerdo a sus estándares de desempeño, o porque considera que es de utilidad.





Una vez que se han definido todos los aspectos anteriores, el área de servicios de información administrativa (MIS) elabora los reportes y los envía por correo al área de mercadotecnia, ellos a su vez los hacen llegar al cliente con la periodicidad que se haya acordado (diariamente, cada tercer día, semanal, quincenal, mensual o anual; y en algunas ocasiones con periodicidad variable).

### **1.5 COSTOS DE OPERACIÓN**

Los costos de operación se generan directamente por el uso del equipo y de las instalaciones, por lo que se generan los siguientes costos de servicios:

#### **Reparaciones mayores**

Son todas aquellas reparaciones tanto del equipo como de las instalaciones que deben realizarse en el taller, de donde se desprenden los diferentes costos:

- Mano de obra. Personal que atenderá la falla.
- Repuestos. Cambio de componentes para el buen funcionamiento del equipo e instalaciones.
- Equipo. Equipo utilizado para una adecuada reparación.

#### **Reparaciones de servicio diario**

Son todas aquellas reparaciones menores que se realizan en el lugar de trabajo y el servicio diario que requiere cada equipo.

#### **Comportamiento del costo**

Conocer cómo se comporta el costo en volúmenes de la producción, resulta esencial en la planificación y el control. El comportamiento del costo plantea un aspecto práctico: al aumentar o disminuir el servicio (nivel de actividad) en un

---



centro de trabajo con cada gasto asignado a ese centro. Los costos pueden clasificarse en dos categorías:

- Costo fijo: Es el que se mantiene constante sin importar el nivel de servicio, por ejemplo: equipo, seguros e impuestos. Los costos fijos pueden ser expresados en costos anualizados.
- Costo variable: Este tipo de costos varía en proporción a los niveles de servicio alcanzado, como ejemplo de este tipo de costo se tiene la mano de obra directa, equipo, etc.

### **Diferencia entre costo y gasto**

Para propósitos de contabilidad, el costo se define como un desembolso que se registra en su totalidad como un activo y se convierte en gasto cuando rinde sus beneficios en el futuro. Por consiguiente, una cuenta de costo es una cuenta de activo.

El gasto se define como un desembolso que se consume corrientemente, o como un costo que ha rendido ya su beneficio. Para fines de la contabilidad, estos términos no se definen con tal rigidez, pues a veces se utilizan para significar un activo y en otras ocasiones un gasto.

### **Gastos indirectos**

Son aquella parte del costo total de producción que no es directamente identificable o rastreable hacia productos o trabajos específicos, los gastos indirectos se componen de:

- Equipo indirecto
- Mano de obra indirecta



### **Gastos de servicio**

Son aquellos que se originan en su manufactura en el departamento de producción, como ejemplo de esta categoría están: los impuestos, los seguros, la depreciación, los suministros generales, los servicios públicos (energía, agua, teléfono, etc.), las reparaciones y mantenimiento entre otros.

Los gastos en términos generales, pueden clasificarse para efectos de planeación en tres grandes agregados:

- Gastos fijos. Son aquellos que mes a mes se mantienen constantes en su monto total, independientemente del volumen de trabajo realizado. Todo esto es válido a un nivel relevante de producción, como son impuestos, seguros, etc.
- Gastos variables. Son aquellos que cambian en su monto total, directamente con las fluctuaciones en la producción o volumen de trabajo realizado. Ejemplos: equipo, mano de obra directa, consumo de energía eléctrica, teléfono, etc.
- Gastos semivariables. Son aquellos que reúnen las dos características anteriores, es decir, que son fijos y variables a la vez, al cambiar la producción o servicios, los gastos semivariables cambian en la misma dirección, pero no en proporción a la fluctuación en la producción. Sin embargo para efectos presupuestales y de planeación financiera deben ser separados en sus componentes.

Una aproximación al cálculo de los gastos indirectos puede plantearse de acuerdo al siguiente esquema, pero esto depende más de los objetivos de control del área encargada. (Ver tabla 1.5.1).

La diferenciación entre gastos controlables y no controlables está íntimamente relacionada con la clasificación de gastos por áreas de trabajo. Los gastos

---



controlables son aquellos que se encuentran sujetos a la autoridad y responsabilidad de un gerente específico.

TIPO DE GASTO	COMPONENTE		
	FIJO	VARIABLE	TOTAL
Sueldos de supervisión	X	X	X
Reparaciones y mantenimiento	X	X	X
Repuestos	X	X	X
Seguros	X	X	X
Impuestos	X	X	X
Salarios	X	X	X
Calefacción y luz	X	X	X
Agua	X	X	X
Teléfono	X	X	X
TOTAL	\$ gasto	\$ gasto	\$ gasto

Tabla 1.5.1 Tipos de gastos indirectos

### Reducción de costos y control de costos

Un programa de reducción de costos (o de gastos) se dirige a esfuerzos específicos para reducir los costos por medio de mejorar los métodos, predisponer el flujo de trabajo y simplificar los servicios

### Planificación de gastos

Al desarrollar un plan táctico de utilidades, se deben evaluar con cuidado los gastos para cada área o centro de trabajo. En armonía con el concepto de la participación, la planificación de gastos debe involucrar a todos los niveles de la administración. La participación es esencial en el desarrollo de presupuestos realistas de gastos para cada área de trabajo.



### **Mantenimiento de instalaciones**

El gasto en el mantenimiento de las instalaciones es variable ya que se presentan en una determinada época o circunstancia, como puede ser la ampliación del edificio o las modificaciones necesarias para el buen desempeño de los grupos de trabajo.

# **CAPÍTULO 2**

## **TEORÍA BÁSICA**



## 2.1 METODOLOGÍAS DE BASES DE DATOS RELACIONALES

A continuación se presenta una breve descripción de las metodologías más representativas en la historia de la ingeniería del software.

### El enfoque entidad-relación de Chen

El modelo entidad-relación (E-R) es el modelo conceptual más utilizado para el diseño de bases de datos. Fue introducido por Peter Chen en 1976. Este modelo está formado por un conjunto de conceptos que permiten describir la realidad mediante representaciones gráficas y lingüísticas:

**Entidad.** Cualquier tipo de objeto o concepto sobre el que se recoge información: cosa, persona, concepto abstracto o suceso. Las entidades se representan gráficamente mediante rectángulos y su nombre aparece en el interior. Un nombre de entidad sólo puede aparecer una vez en el esquema conceptual.

Hay dos tipos de entidades: fuertes y débiles. Una entidad débil es una entidad cuya existencia depende de la existencia de otra entidad. Una entidad fuerte es una entidad que no es débil.

**Relación.** Es una correspondencia o asociación entre dos o más entidades. Cada relación tiene un nombre que describe su función. Las relaciones se representan gráficamente mediante rombos y su nombre aparece en el interior.

**Atributo.** Es una característica de interés o un hecho sobre una entidad o sobre una relación. Los atributos representan las propiedades básicas de las entidades y de las relaciones. Toda la información extensiva es portada por los atributos. Gráficamente, se representan mediante círculos que salen de las entidades o relaciones a las que pertenecen.



La definición del modelo conceptual con la técnica propuesta por Chen propone una secuencia de fases para la obtención del modelo:

- Identificar las entidades dentro del sistema: para ello debe conocerse el funcionamiento del sistema en estudio, a través de estudios de usuarios, de necesidades de información, de tipos de información, etc. Como guía puede utilizarse para la definición de las entidades objetos reales, personas, actividades del sistema, objetos abstractos, etc.
- Determinar las claves o identificadores de entidades: señalar aquellos atributos que identifiquen inequívocamente cada ocurrencia de la entidad, y que no puedan ofrecer valores nulos.
- Establecer las relaciones entre las entidades describiendo el grado de las mismas: estudiar las asociaciones entre entidades para definir su importancia dentro del contexto del sistema y obtener su cardinalidad.
- Dibujar el modelo de datos: representar gráficamente el modelo obtenido. Identificar y describir los atributos de cada entidad: señalar aquellas propiedades de la entidad de interés para el sistema.
- Verificaciones: eliminación de las relaciones redundantes y que puedan ser obtenidas a través de combinar otras asociaciones.

### **Transformación en un modelo relacional**

Las entidades y sus atributos se representan físicamente a través de tablas en las que los datos se almacenan en dos dimensiones. Las filas de la tabla contienen los atributos de cada una de las entidades, y las columnas el conjunto de atributos del mismo tipo de cada entidad. El grado de la tabla corresponderá al número de columnas de la tabla. En este momento estaremos trasladando el modelo entidad-relación al modelo relacional, es decir, la transformación entre el modelo conceptual y el lógico.





Uno de los puntos fuertes de este modelo, es que prevé que las entidades puedan mantener relaciones entre ellas. En primer lugar, es necesario definir la llave de la entidad. Las llaves serán el atributo, o conjunto de atributos, pertenecientes al mismo tipo de entidad que hacen único el acceso a esa entidad u ocurrencia de la tabla, determinando de esta forma a una única entidad.

La presencia de varios atributos que pueden funcionar como llave, dan lugar a la existencia de llaves candidatas, y por otra parte se puede hablar de llaves simples (formadas por un único atributo) y múltiples, compuestas o concatenadas (formadas por un conjunto de atributos). No hay que obviar tampoco el concepto de llave foránea, aquel atributo de una tabla que puede funcionar como llave en otra tabla. La ocurrencia de entidad será, cada uno de los posibles valores reales que puede tomar la llave de una entidad.

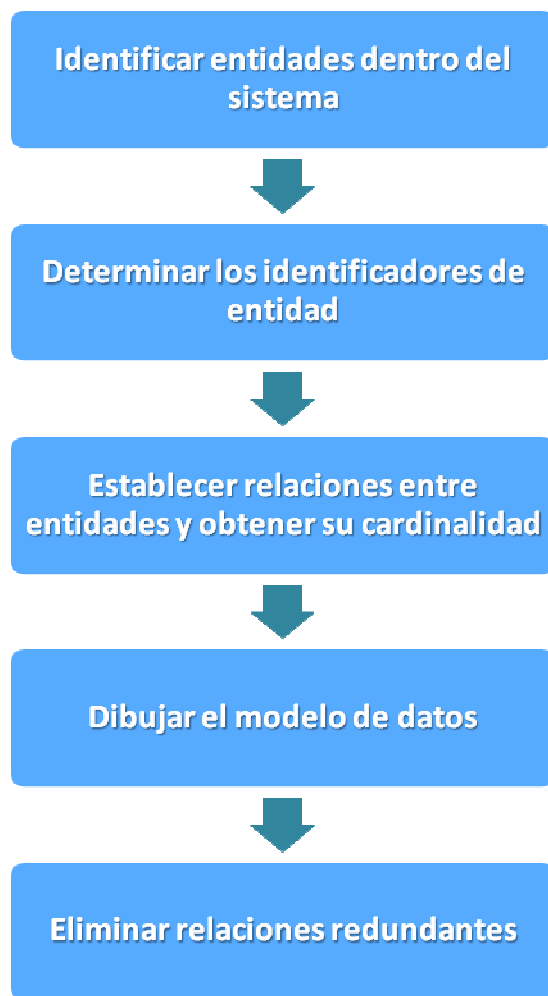
Las relaciones entre tablas, basadas en la conexión de éstas a través de las llaves, pueden ofrecer diferentes cardinalidades, entendiendo como tal el número de ocurrencias de una entidad que se relacionan con ocurrencias de otra entidad. Pueden identificarse tres tipos: (1,1), donde una ocurrencia se relaciona con otra; (1, m), donde una ocurrencia puede relacionarse con varias; y (m,m), donde varias ocurrencias de una entidad pueden relacionarse con varias ocurrencias de otra entidad. El modelo de Chen es n-ario, lo cual quiere decir que las relaciones pueden establecerse entre una, dos o más entidades.

El modelo conceptual de datos obtenido mediante la técnica de entidad/relación será refinado y convertido en un modelo lógico relacional, utilizando la normalización, lo que ofrecerá como resultado el conjunto de tablas a implementar en la base de datos. Su finalidad es reducir las inconsistencias y redundancias de los datos, facilitar el mantenimiento y evitar las anomalías en las manipulaciones de datos. El objetivo será obtener un modelo lógico normalizado que represente las entidades normalizadas y las interrelaciones existentes entre ellas.

---



En la Figura 2.1.1 se muestran los pasos a seguir en el enfoque entidad-relación de Chen.



**Figura 2.1.1 Enfoque entidad-relación de Chen**

### **El Análisis Estructurado de Yourdon / De Marco**

Para facilitar el uso de sistemas informáticos complejos en la ejecución de proyectos, Ed Yourdon ideó una metodología de análisis conocida como análisis

---



estructurado cuya característica fundamental consistía en que el objeto principal del análisis es la información que se intercambia entre procesos, a diferencia del enfoque tradicional que se centraba en los procesos propiamente dichos.

Posteriormente, Tom De Marco desarrolló y aplicó con excelentes resultados esta metodología. Desde el punto de vista de su aplicación, las tareas básicas del análisis estructurado son las siguientes:

- Estudio del entorno físico existente. Esta tarea debe enfocarse desde el punto de vista de los datos o información que fluye entre procesos, más que desde el punto de vista de las personas ejecutoras. El resultado final es un modelo lógico sin relaciones físicas concretas.
- Desarrollo del nuevo modelo lógico propuesto, por medio de los diagramas de flujo de datos (DFD's) correspondientes, el diccionario de datos (que, explican el contenido de los flujos de datos o información entre procesos) y las descripciones de procesos (que explican el contenido de las actividades o procesos propiamente dichos).
- Cuando la complejidad de la actividad lo hace necesario, se desarrollan modelos lógicos alternativos, evaluándolos mediante los adecuados estudios comparativos y seleccionando la solución óptima.
- Elaboración de los documentos finales de los resultados.

Los elementos claves del análisis son los DFD's que se elaboran con los siguientes elementos básicos: (Ver Figura 2.1.2)

- Vectores: representan los flujos de datos. La mayor parte de los datos se mueven entre procesos, pero también pueden hacerlo desde o hacia los archivos y desde o hacia personas o entidades. En los DFD's los vectores se identifican con la denominación de la información que se mueve por ellos.



- Burbujas: representan los procesos, que consisten en la transformación de información. En los diagramas se asigna un índice a cada burbuja.
- Líneas rectas paralelas: representan los almacenamientos de datos. Los archivos son almacenes temporales de datos. Cualquier elemento que sirva para almacenar información se considera un archivo y en los DFD's se representa por unas líneas rectas paralelas con su identificación incorporada.
- Cajas: representan las entidades externas, emisores o receptores de información, normalmente personas o entidades, pertenecientes al ámbito del sistema. Suelen representar elementos del entorno del problema principal. Se incluyen únicamente como referencia del contexto físico exterior.

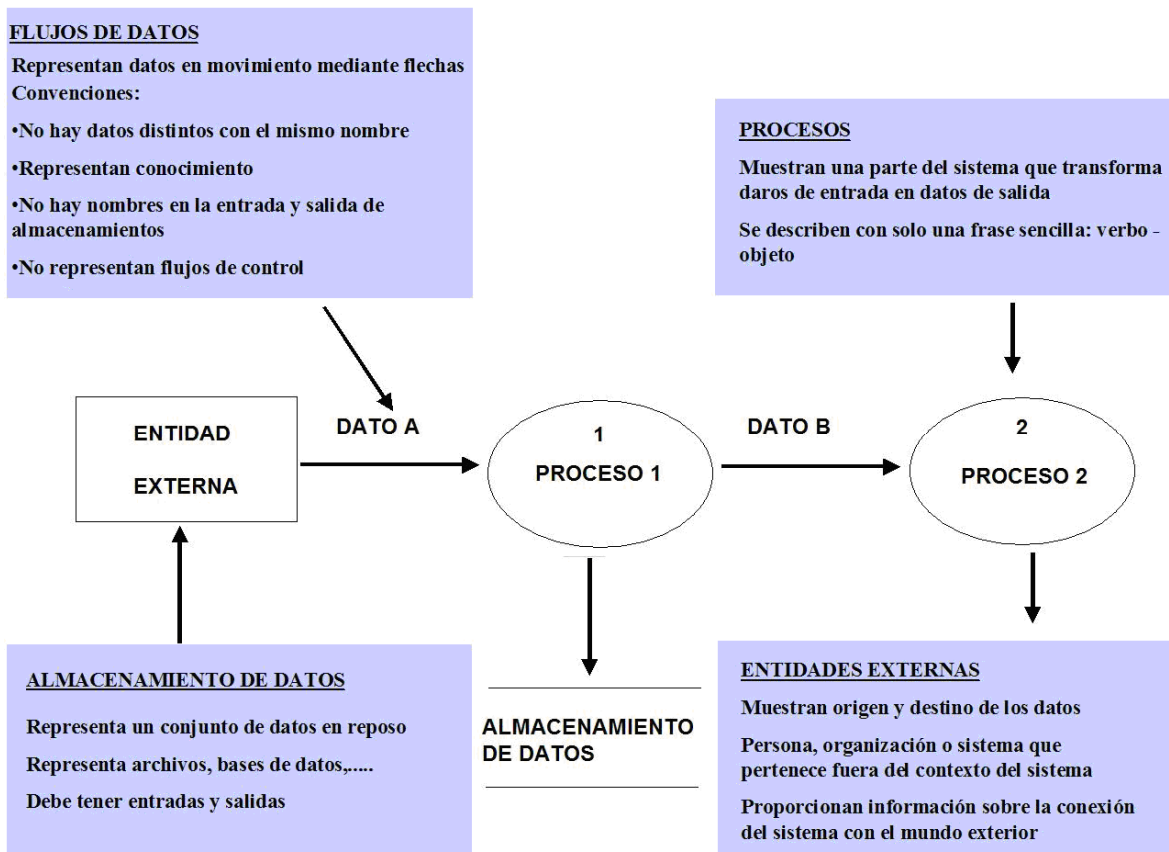
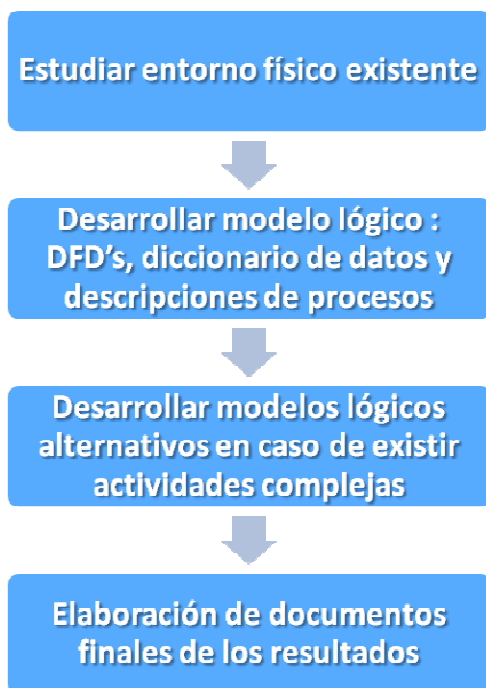


Figura 2.1.2 Diagrama de Flujo de Datos (DFD)



La Figura 2.1.3 muestra los pasos que se deben seguir en el análisis estructurado Yourdon / De Marco.



**Figura 2.1.3 Análisis estructurado Yourdon / De Marco**

### **Metodología MERISE**

Esta metodología comenzó a ser desarrollada en 1972 por un equipo universitario. El proyecto finalizó en mayo de 1978 dando lugar a MERISE como metodología de Análisis y Diseño de Sistemas de Información.

Al desarrollar una aplicación se tienen que considerar los tres ciclos siguientes: Abstracción, Decisión, Vida.



### Ciclo de abstracción

Se basa en la percepción de tres niveles de abstracción: conceptual, organizativo y físico. Además se definen dos niveles para cada nivel un modelo de datos y otro de tratamientos. (Ver Tabla 2.1.1).

Nivel de abstracción	Tratamientos	Datos
Conceptual	Modelo conceptual: Indica lo que el sistema tiene que resolver	Modelo conceptual: Trabaja con objetos, entidades, describe propiedades
Organizativo	Modelo organizativo: Indica quién lo hace, cuándo y dónde	Modelo lógico: Transforma el modelo conceptual en estructural, lógicas de datos.
Técnico	Modelo operativo: Indica cómo se hace.	Modelo físico: Se concreta la estructura final en función del gestor a utilizar.

**Tabla 2.1.1 Ciclo de abstracción**

### Ciclo de decisión

Estaría formado por los analistas y los órganos directivos de la empresa. Son los que plantean el sistema, los que dicen lo que hay que hacer y toman la decisión de hacer o no hacer el proyecto.

Están todos los recursos humanos con los que cuenta la empresa. Se recoge una serie de información para tomar decisiones. Los programadores con un jefe de equipo que coordina los trabajos elaboran una estrategia de desarrollo. Tiene que haber un representante de la empresa a la que se le está desarrollando la aplicación que supervise lo anterior.



### Ciclo de vida

Son todos los procesos tanto de construcción como de ejecución que se llevan a cabo en una aplicación, desde que se concibe la idea de realizarla hasta que se deja de utilizar. Se aporta un ciclo de vida más largo a los existentes hasta que se materializa en un conjunto definido de etapas.

Las fases de la metodología MERISE se muestran en la Figura 2.1.4:



**Figura 2.1.4 Fases de la Metodología MERISE**

### Metodología SSADM (Structured Systems Analysis and Design Method)

La Metodología de Análisis y Diseño de Sistemas Estructurados (SSADM) en común con otras metodologías estructuradas, adopta un acercamiento al desarrollo de sistemas de información en el cual especifica por adelantado

---



módulos, etapas y tareas que se deben realizar, los entregables que deben ser producidos y las técnicas usadas para producir los entregables.

SSADM adopta el modelo de cascada de desarrollo de sistemas, donde cada fase tiene que ser terminada antes de que las fases subsecuentes puedan comenzar. (Ver Figura 2.1.5).

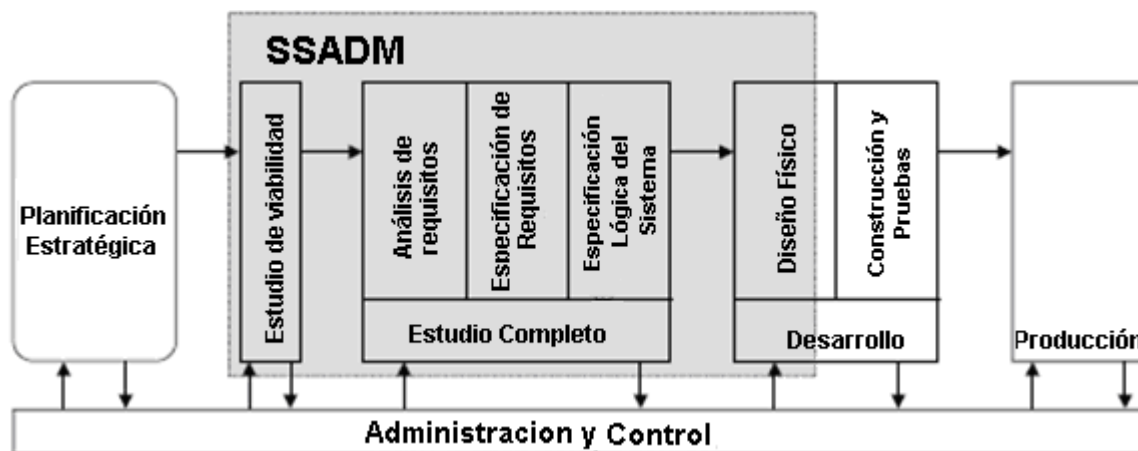


Figura 2.1.5 Metodología SSADM

El método SSADM implica el uso de una secuencia de análisis, documentación y tareas de diseño:

### Etapa de viabilidad

Analiza la situación actual a nivel global. Un DFD (Diagrama de Flujo de Datos) es usado para describir como trabaja el sistema actual y así poder visualizar problemas conocidos. Los pasos siguientes son parte de esta etapa:

- Desarrollo del Modelo de Actividad de Negocio. Un modelo de actividad de negocio es construido. Eventos y reglas de negocio pueden ser examinados como entradas para la especificación de la automatización del nuevo sistema





- Examinar y definir requerimientos. El objetivo de este paso es identificar problemas asociados con la situación actual y que deben ser resueltos por el nuevo sistema. Esto puede ayudar a identificar servicios adicionales para ser proporcionados por el nuevo sistema
- Examinar el proceso actual. Este paso sirve para examinar el flujo de la información asociada con los servicios proporcionados actualmente y los describe en forma de Modelo de Flujo de Datos. En este punto, el Modelo de Flujo de Datos representa los servicios actuales con todas sus deficiencias. No se deben incorporar mejoras requeridas
- Examinar los datos actuales. Este paso es para identificar y describir la estructura de los datos del sistema, independientemente de la manera en la que los datos estén almacenados y organizados actualmente. Esto produce un modelo de datos que respalda los servicios actuales
- Obtener la vista lógica de los servicios actuales. El objetivo de este paso es desarrollar una vista lógica del sistema actual que puede ser usada para entender problemas existentes.

### **Especificación lógica del sistema**

Esta etapa se divide en dos partes: La primera parte investiga el ambiente existente. En esta parte, los requerimientos del sistema son identificados y la situación actual de negocio es modelada. El modelado consiste en la creación de un DFD (Diagrama de flujo de datos) y LDS (Estructura Lógica de Datos) para procesos y estructuras de datos que forman parte del sistema. En la segunda parte, seis opciones de negocio son presentadas (BSOs). Una de éstas debe ser seleccionada y construida.

### **Especificación de requerimientos**

Para ayudar a seleccionar la mejor opción, un número de opciones de sistema de negocio deben describir el alcance y la funcionalidad proporcionada por un

---



desarrollo en particular, debiendo ser preparada y presentada. Estas opciones pueden ser respaldadas por la documentación técnica como el LDM (el Modelo Lógico de Datos) y el DFD. También se tiene que preparar una evaluación financiera que debe estar sustentada por descripciones de implementación.

### **Etapas de especificación lógica del sistema**

En esta etapa las opciones técnicas de viabilidad son seleccionadas. El ambiente de desarrollo / implementación es especificado basados en esta opción.

### **Diseño lógico del proceso**

El objetivo de esta etapa es especificar los datos físicos y procesos que serán actualizados. Adicionalmente se especificarán los diálogos. Se tienen que seguir los pasos siguientes:

- Definir diálogo de usuario. Este paso define la estructura de cada diálogo requerido para soporte de funciones en línea e identificadores de requerimientos de navegación, ambos dentro del diálogo y entre diálogos
- Definir procesos de actualización. Para completar la especificación de la actualización de la base de datos requerida por cada evento y para definir el manejo de errores para cada evento
- Definir procesos de consulta. Para completar la especificación de los procesos de consulta de la base de datos y para definir manejo de errores para cada consulta

### **Diseño físico**

El objetivo de esta etapa es especificar los datos físicos y el diseño de procesos usando el lenguaje y características del ambiente físico seleccionado e incorporando normas de instalación. Las actividades siguientes son parte de esta etapa:

---



- Prepararse para el diseño físico
- Aprender las reglas del ambiente de desarrollo
- Repasar los requerimientos exactos para el mapeo lógico y físico
- Planear el acercamiento
- Completar la especificación de funciones
- En forma incremental y repetidamente desarrollar los diseños de procesos y datos

Las técnicas más importantes usadas en SSADM son:

- Modelo Lógico de Datos. Es el proceso de identificación, modelado y documentación de los requerimientos de datos del sistema que está siendo diseñado. Los datos son separados en entidades (cosas sobre las cuales un negocio necesita guardar información) y relaciones (las asociaciones entre las entidades).
- Modelo de Flujo de Datos. Es el proceso de identificación, modelado y documentación de cómo se mueven los datos alrededor de un sistema de información. Examina procesos (las actividades que transforman los datos de una forma a otra), almacenamiento de datos (las áreas para almacenar los datos), entidades externas (que datos son enviados o recibidos de un sistema) y los flujos de datos (rutas por las cuales los datos pueden fluir).
- Modelo de Comportamiento de Entidades. Este es el proceso de identificación, modelado y documentación de los acontecimientos que afectan cada entidad y secuencia en la cual estos eventos ocurren.

Nota: Esta metodología no construye ni prueba, sólo ayuda al planteamiento del problema. (Ver Figura 2.1.6).

---

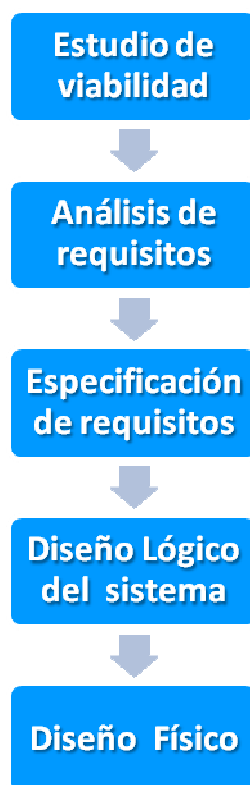


Figura 2.1.6 Metodología SSADM

## 2.2 BASES DE DATOS Y CARACTERISTICAS GENERALES DE LOS SITIOS WEB

A continuación se presentan algunas definiciones de una base de datos:

- Es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso.
- Es el conjunto de información organizada (archivo lógico, en uno o varios archivos físicos).



La finalidad de una base de datos es servir a una aplicación o más, de la mejor manera posible. Existen varios niveles en que puede observarse la base de datos, conocidos como abstracción de la información (Ver Figura 2.2.1).

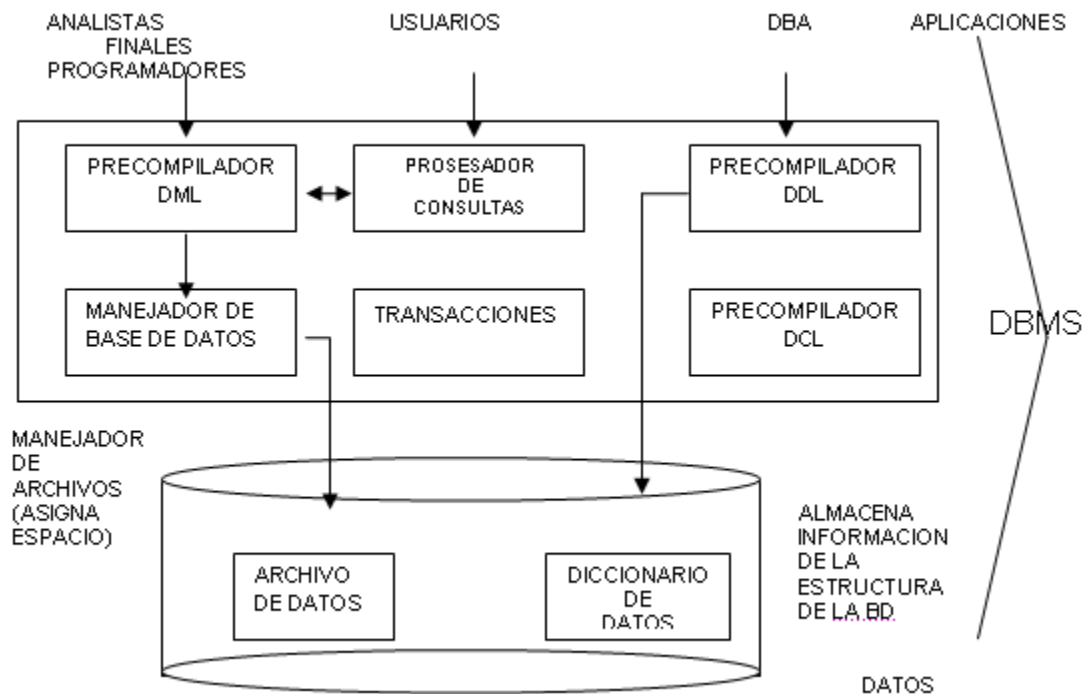


Figura 2.2.1 Estructura de la base de datos

- Nivel físico (Hardware). Describe como se almacenan los datos (bytes)  
DBA = Administrador de la base de datos
- Nivel conceptual (Analista). Esquematización de los datos y sus relaciones (diagramas) analista y desarrollador
- Nivel de visión. Abstracción más alta, como interpretan la información los usuarios finales



### **Ventajas de un sistema de base de datos**

- Independencia de los datos. Podemos definirla como la independencia de la representación de la información respecto a las aplicaciones que la utilizan. De esta forma, se consigue una representación conveniente para todos los usos posibles de los datos y la estandarización de procedimientos.
- Distintas aplicaciones necesitan distintas 'vistas' de los datos.
- Es posible modificar la estructura de almacenamiento de la información sin afectar a las aplicaciones que los utilizan.

**Reducción de la redundancia.** Evita el almacenamiento múltiple de una misma información para uso de distintas aplicaciones, o en distintos departamentos con propósitos diferentes. Como veremos, además de la economía importante en coste de mantenimiento de la información y la posibilidad de extender el uso de la información, se consigue también evitar algunos problemas que puede producir la redundancia.

**Evitar inconsistencias.** Impide que exista información discrepante sobre un mismo y único hecho. La aparición de información inconsistente e incluso contradictoria puede darse si se almacena redundantemente información relativa a un mismo hecho u objeto.

**Compartir datos.** Permite utilizar los mismos datos entre distintos usuarios y aplicaciones, gestionando el acceso concurrente de todas ellas a la información.

**Garantizar la seguridad.** Permite garantizar la seguridad de la información, controlando el acceso y la manipulación de la información por las distintas aplicaciones y usuarios. También mantiene la integridad de la información.



**Balancear aplicaciones conflictivas.** Permite balancear la utilización de los recursos existentes, en capacidad de almacenamiento y de procesamiento entre las necesidades de los distintos usuarios y aplicaciones.

### **Desventaja de un sistema de base de datos**

La seguridad y la integridad pueden ser contraproducentes sin buenos controles.

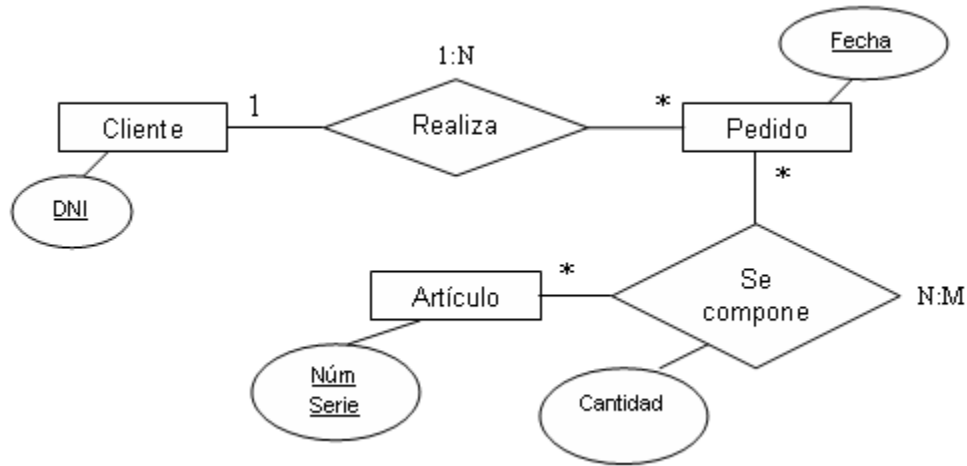
### **Base de datos relacional**

Una base de datos relacional es un conjunto de dos o más tablas estructuradas en registros (líneas) y campos (columnas), que se vinculan entre sí por un campo en común, en ambos casos posee las mismas características como por ejemplo el nombre de campo, tipo y longitud; a este campo generalmente se le denomina ID, identificador o clave. A esta manera de construir bases de datos se le denomina modelo relacional.

El Modelo Entidad/Relación (E/R) es un concepto de modelado para bases de datos, el cual pretende visualizar los objetos que pertenecen a la base de datos como entidades, las cuales tienen atributos y se vinculan mediante relaciones. Es una representación lógica de la información. Mediante una serie de procedimientos se puede pasar del modelo E/R a otros, como por ejemplo el modelo relacional.

Los diagramas entidad-relación (denominado por sus siglas, E/R) son una herramienta para el modelado de datos de un sistema de información.

Estos diagramas expresan entidades relevantes para un sistema de información, sus Interrelaciones y propiedades (Ver Figura 2.2.2).



**Figura 2.2.2 Diagrama del modelo entidad / relación**

Las bases de datos relacionales son normalizadas para tres puntos en específico:

- Evitar la redundancia de los datos.
- Evitar problemas de actualización de los datos en las tablas.
- Proteger la integridad de los datos.

En el modelo relacional, es frecuente llamar tabla a una relación, aunque para que una tabla bidimensional sea considerada como una relación, tiene que cumplir con algunas restricciones:

- Cada columna debe tener su nombre único.
- No puede haber dos filas iguales. No se permiten los duplicados.
- Todos los datos en una columna deben ser del mismo tipo.





Las bases de datos relacionales pasan por un proceso al que se le conoce como normalización de una base de datos, la cual es entendida como el proceso necesario para que una base de datos sea utilizada de manera óptima.

### **Normalización de una base de datos**

Se basa en describir la información usando tablas. Estas tablas se intentan estructurar de forma que cumplan unos formatos llamados Formas Normales. Cuanto más alta la forma normal, más estrictos son los criterios que cumple la tabla y más fácil resulta tratarla.

- Primera FN: No hay campos múltiples
- Segunda FN: Cada atributo que no forme parte de la clave primaria mantiene una dependencia funcional total respecto a la clave primaria (no depende funcionalmente de un subconjunto de la clave primaria).
- Tercera FN: No hay dependencias transitivas
- Tercera FN de Boyce-Codd: No hay más de una clave primaria que determine funcionalmente (de forma redundante) algún atributo.
- Cuarta FN y Quinta FN: No se describen porque no suelen utilizarse.

### **Álgebra relacional**

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

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



### Las operaciones básicas

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

#### Selección ( $\sigma$ )

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

$$\sigma_F(R)$$

Una condición puede ser una combinación booleana, donde se pueden usar operadores como:  $\wedge$ ,  $\vee$ , combinándolos con operadores  $<$ ,  $>$ ,  $\leq$ ,  $\geq$ ,  $=$ ,  $\neq$ .

#### Proyección ( $\Pi$ )

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

$$\Pi_{A,B,\dots}(R)$$

Donde A y B son atributos de la relación R.

#### Producto cartesiano ( $\times$ )

El producto cartesiano de dos relaciones se escribe como:

$$R \times S$$

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

#### Unión ( $\cup$ )

La operación  $R \cup S$  retorna el conjunto de tuplas que están en R, o en S, o en ambas. R y S deben ser uniones compatibles.



### Diferencia (-)

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

$$R - S$$

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

Estas operaciones son fundamentales en el sentido en que:

- Todas las demás operaciones pueden ser expresadas como una combinación de éstas.
- Ninguna de estas operaciones pueden ser omitidas sin que con ello se pierda información.

### Las operaciones no básicas

Entre los operadores no básicos tenemos:

#### Intersección ( $\cap$ )

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

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

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

#### Join ( $\Join$ )

El join de dos relaciones es equivalente a:

$$R \Join_F S = \sigma_F (R \times S)$$

Esto es mucho más útil que el uso del operador básico producto cartesiano, pues especifica una regla para la combinación de los atributos.



### División (/)

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

El operador división  $A / B$  retorna todos los distintos valores de “ $x$ ” tales que para todo valor “ $y$ ” en  $B$  existe una tupla  $\langle x, y \rangle$  en  $A$ .

El **cálculo relacional** es un lenguaje de consulta que describe la respuesta deseada sobre una base de datos sin especificar como obtenerla, el cálculo relacional es de tipo declarativo; pero siempre ambos métodos logran los mismos resultados.

### Características generales de los sitios web

Un sitio web (website) es un conjunto de páginas web, típicamente comunes a un dominio de internet o subdominio en la world wide web (www) en internet. Una página web es una fuente de información adaptada para la world wide web (www) y accesible mediante un navegador de internet. Ésta información se presenta generalmente en formato html y puede contener vínculos a otras páginas web, constituyendo la red enlazada de la world wide web.

Dichas páginas pueden ser cargadas por un servidor local o remoto, llamado servidor web. El servidor web puede restringir las páginas a una red privada, por ejemplo, una intranet, no puede publicar las páginas en la world wide web. Las páginas web son solicitadas y transferidas de los servidores usando el **protocolo de transferencia de hipertexto (HTTP- Hypertext Transfer Protocol)** y pueden consistir en archivos de texto estático, o se pueden leer una serie de archivos con código que instruyan al servidor cómo construir el html para cada página que es solicitada, a esto se le conoce como página web dinámica.



El dominio de internet es un nombre de equipo que proporciona nombres más fácilmente recordados en lugar de la IP numérica. Permiten a cualquier servicio moverse a otro lugar diferente en la topología de internet, que tendrá una dirección IP diferente.

Los sitios web deben contar con enlaces para permitir navegar por la web más fácilmente. Se pueden utilizar teclas de acceso para facilitar la navegación. Para facilitar la orientación de los visitantes, se puede colocar un pequeño menú de seguimiento que sitúa la página en la que se está navegando con respecto a la inicial.

En los sitios web se pueden encontrar dos tipos distintos de ligas: las ligas externas, que enlazan a otro sitio web y las ligas internas, que enlazan a otras partes del sitio web. Las ligas externas se abren en una ventana nueva. Las ligas internas se abren en la ventana activa.

Los sitios web se pueden escribir con diferentes lenguajes como xhtml, CSS y JavaScript, y para asegurar que funciona correctamente en todos los dispositivos de internet, el código debe ser validado por el W3C. Al mismo tiempo, se debe diseñar de tal forma que funcione también en dispositivos que no soporten CSS o JavaScript.

### **Características de un sitio web efectivo**

- Contar con información actualizada. Presentando a los visitantes información reciente del sitio, creando una imagen profesional y confiable.
- Tener información completa. Tratar de no mostrar información a medias, diseño web efectivo.



- Presentación adecuada de la información. El diseño agradable acompañado de una estructura de navegación eficiente, dará a los visitantes del sitio una buena experiencia.
- Presencia en buscadores. Es completamente inútil tener el mejor de los sitios web si no está presente en los buscadores de internet.

### **Fácil integración**

La integración del contenido es tan sencilla como colocar un hipervínculo en el sitio web del cliente que llama a un servidor, quien una vez comprobados una serie de parámetros devuelve el contenido demandado con el formato definido para el sitio web.

### **Escalabilidad**

Este servicio está pensado para que pueda acceder exclusivamente a aquellos módulos de contenido que le interesan, de tal modo que puede crear, utilizando los contenidos, secciones a la carta dentro del sitio web.

### **Personalización**

La tecnología desarrollada permite personalizar, para cada sitio web, una amplia hoja de estilos, lo que facilita la integración visual de los contenidos en otros sitios web, de tal modo que el usuario del sitio web no tenga la sensación de que ha cambiado de página web al visualizar los contenidos.

### **Características técnicas de la web**

- El diseño y composición de las páginas se realiza mediante el sistema de programación por capas.
- El sitio web facilita la descarga de páginas y el ahorro de proceso gracias a la utilización de un sistema de memoria caché del navegador.
- Todo el contenido de las páginas se visualiza en la misma ventana, sin utilizar marcos.



- Toda la maquetación de las páginas se desarrolla utilizando **hojas de estilo en cascada css (Cascading Style Sheets)** utilizadas para definir la presentación de un documento html o xml.
- Las hojas de estilo css facilitan la uniformidad de la web y la aplicación de técnicas de accesibilidad tales como el cambio de tamaño del texto desde el propio navegador.
- Para visualizar correctamente el sitio web se necesita una resolución de pantalla mínima de 800x600 con 256 colores. Para que sea óptimo se necesita una resolución de 1024x768 y una calidad de colores de 32 bits, que es la opción recomendable.

### **Características del los sitios web**

Hay muchos nombres de dominio que son registrados con el solo propósito de reservar el nombre para usarlo después. Y sólo 1/4 de los sitios web son sitios con al menos dos páginas. Observamos que el 31% se trata de sitios con la navegación construida usando Flash o JavaScript, el 36% se trata de sitios que redirigen al usuario a otro sitio usando comandos html que reenvían al browser a otra dirección; esto es usual cuando una organización tiene un sitio web con un nombre largo, y utiliza otros nombres cortos para mejorar la visibilidad de sus sitios y el 33% restante efectivamente sólo tiene una imagen o un poco de texto, sin más información que eso.

### **Tamaño de las páginas en un sitio web completo**

El tamaño promedio de un sitio web completo, considerando solamente las páginas html, es de aproximadamente 0.8 megabytes. Esto es sólo una pequeña fracción de la información total, puesto que las imágenes constituyen una parte importante del tamaño de las páginas.



### **Profundidad máxima**

La página inicial de un sitio web tiene profundidad 0, y el nivel de cada página siguiente es el número de clicks que son necesarios para llegar a esa página.

### **Dominios y direcciones IP**

Los dominios con más sitios son los proveedores de internet y universidades, quedando más atrás el gobierno, donde la variante en inglés es más popular que el castellano. En promedio, cada dirección IP que tiene un servidor web atiende a 3.4 sitios distintos. Si bien, la mayoría de los sitios tiene su propia dirección IP, existen más de 90 direcciones IP que tienen más de 100 sitios cada una, incluso 2 direcciones IP con más de 1000 sitios cada una.

### **Enlaces**

Una característica central de las páginas web es que pueden contener enlaces o hipervínculos a otras páginas. En el caso de la web, estos enlaces conforman una estructura que se denomina red libre de escala. Las redes libres de escala, al contrario de las redes aleatorias, se caracterizan por una distribución dispareja de enlaces y se caracterizan por tener unos pocos nodos altamente enlazados que actúan como centros que conectan muchos de los otros nodos a la red.

## **2.3 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE ASP**

### **Características de ASP**

**Active Server Page (ASP)** es una tecnología desarrollada por Microsoft que se utiliza para crear páginas Web dinámicas, teniendo como características principales la sencillez de su uso y la integración con otros productos Microsoft, la facilidad de acceso a bases de datos vía **ADO (Access Data Object)**, la





posibilidad de utilización de multitud de objetos integrados y no integrados así como la eficiencia en la ejecución.

Las páginas ASP contienen dos partes: código de programación y html incrustado. El código de programación puede escribirse en diferentes lenguajes de secuencias de comandos. Un lenguaje de secuencia de comando es una sintaxis específica que se utiliza para ejecutar instrucciones en una computadora.

Un programa formado por instrucciones de un lenguaje de secuencias de comandos en particular se conoce como archivo de secuencias de comandos. Una página ASP siempre deberá tener la extensión (.asp).

Entre los lenguajes de secuencias de comandos populares para web se encuentran el VBScript y JavaScript. Al crear una página ASP, es posible utilizar uno de los cuatro lenguajes de programación siguientes:

- VBScript. Este es el lenguaje de secuencias de comandos utilizado con mayor frecuencia para las páginas ASP, y su sintaxis es similar a la de Visual Basic.
- Jscript. Parecido al JavaScript.
- PerlScript: Similar al Perl.
- Python: Un poderoso lenguaje de secuencias de comandos utilizado comúnmente para el desarrollo en Web.

### **Modelo cliente / servidor**

En un modelo cliente/servidor, dos computadoras actúan en conjunto para realizar una tarea en la que una computadora cliente solicita información necesaria a una computadora servidor, ésta devuelve la información y el cliente la procesa.



Las paginas ASP contienen una combinación de html y código de programación, este código el cual se puede escribir en diversos lenguajes permite que las paginas ASP sean dinámicas sin embargo el servidor web tiene que procesar este código de programación antes de enviar el html al cliente. Cuando un explorador web solicita una página ASP ocurre lo siguiente:

- El cliente (explorador web) localiza al servidor web especificado en la primera parte de la URL (www.unam.com).
- Después el cliente solicita la página ASP especificada en la segunda parte del URL (/ejemplo.asp).
- El servidor web lee el archivo ASP y procesa el código.
- Una vez que el servidor web ha procesado completamente la página ASP, la salida se envía al cliente en formato html.
- El cliente recibe el html enviado por el servidor y lo interpreta para el usuario.

Cuando el servidor web procesa una página ASP, todo el código de programación es interpretado en el servidor (nada de éste es enviado al cliente). Ver Figura 2.3.1



**Figura 2.3.1 Interacción cliente/servidor para archivos ASP**



No siempre se necesita un servidor web dedicado para publicar páginas de ASP que se han creado. Se puede instalar un software de servidor web en una computadora personal. El software de Personal Web Server y de **Internet Information Services (IIS)** incluyen soporte para ASP. Personal Web Server permite desarrollar un sitio antes de transferirlo a un proveedor de presencia de la web (servidor dedicado). IIS ofrece varias opciones avanzadas que permiten hospedar y administrar efectivamente hasta los sitios mas ocupados en la web.

### **Páginas web activas en el servidor**

Las páginas web activas en el servidor se caracterizan porque los cómputos asociados a las páginas se realizan en el equipo servidor. De esta manera los visualizadores de los clientes reciben únicamente instrucciones html. Por lo tanto la complejidad del sistema se traslada a los equipos servidores liberando la carga de los equipos de los usuarios. Las principales situaciones en las que este mecanismo resulta adecuado son:

- Cuando se necesita acceder a información centralizada (habitualmente situada en bases de datos localizadas en el equipo servidor).
- Cuando los procesos que se ejecutan, por razones de seguridad, no se traspasan a los equipos de los usuarios.
- Cuando no se desea que los usuarios posean el código asociado a la pagina web.
- Cuando se desea estar seguro de que todos los clientes van a poder ver correctamente las páginas web, independientemente del tipo y versión del visualizador y del equipo que posean.



### Aplicaciones ASP

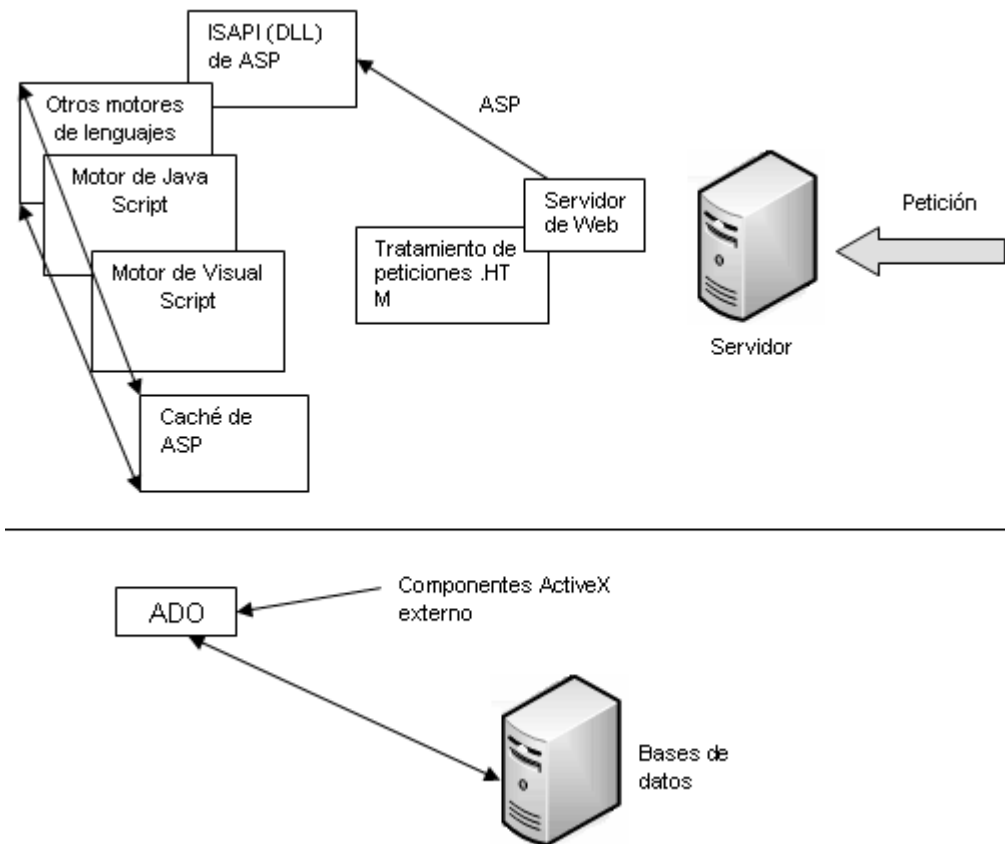
Los desarrollos realizados con ASP no necesitan compilaciones que retarden el proceso de producción, y los errores de programación no provocan la caída del servidor web como es habitual en otro tipo de aplicaciones.

Desde ASP se pueden realizar accesos a componentes ActiveX que se ejecutan en el servidor. De esta manera se hace un uso muy simple de ODBC para el acceso a distintos tipos de bases de datos. ASP proporciona un método eficiente y sencillo de crear sitios web con páginas dinámicas y accesos a bases de datos.

Cuando se trabaja con IIS y ASP, el servidor web analiza las peticiones de página que recibe. Si se encuentra con una solicitud de página con extensión (.asp) en lugar de (.htm) entonces se apoya en la aplicación ISAPI (Internet Server API) la cual es una DLL de Windows que se ejecuta en el mismo espacio de direcciones que el servidor web cuyo código es reentrante y puede soportar las peticiones simultáneas de diversos clientes con una sola imagen en memoria y esta sirve de soporte de ejecución de las páginas ASP.

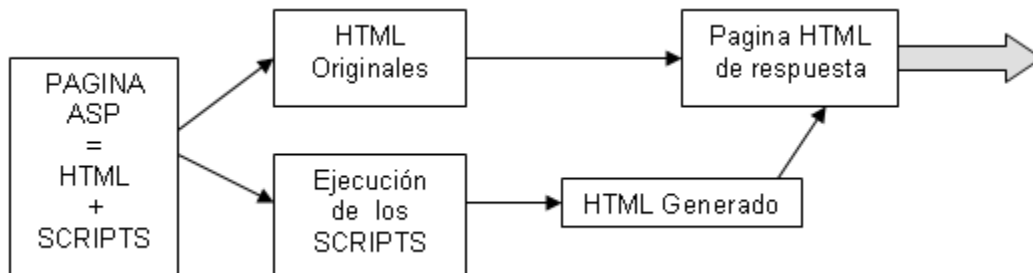
La aplicación ISAPI de ASP, diferencia las líneas de html de las instrucciones que dan la funcionalidad dinámica a las páginas activas. Cuando determina el lenguaje en el cual se encuentran los programas (SCRIPTS), da paso al motor de ejecución de scripts adecuado. Los motores de ejecución de scripts se encargan de realizar el análisis sintáctico y la compilación de las instrucciones ejecutables. Una vez resueltas las fases anteriores se procede a ejecutar las instrucciones.

Los motores de ejecución de scripts a menudo se encuentran con objetos ActiveX exteriores con los que tiene que interactuar. Un ejemplo muy importante de esta situación se centra en el acceso a bases de datos a través de ADO (Access Data Objects) basado en tecnología COM (Component Object Model). Ver Figura 2.3.2



**Figura 2.3.2 Empleo de motores de diversos lenguajes en un entorno ASP**

El usuario recibe como respuesta un archivo (.htm), que se ha formado uniendo las instrucciones html originales de la página (.asp) con las instrucciones html que se ha generado tras la ejecución de los scripts (Ver Figura 2.3.3).



**Figura 2.3.3 Mecanismo de obtención de una página de respuesta con ASP**



### Objetos de ASP

ASP dispone de mecanismos muy sencillos para recoger la información proveniente de un formulario y para enviar al usuario la respuesta procesada por el servidor. En ambos casos se utilizan los objetos integrados que ocultan los detalles más complejos relativos al formato de la información intercambiada.

Existen cinco objetos principales de ASP:

- Response: envía y controla la información del servidor web a un usuario.
- Request: extrae y controla los datos enviados de un usuario al servidor web.
- Server: crea objetos y suministra accesos a métodos y propiedades en el servidor web.
- Session: almacena información de sesión para usuarios individuales conforme navegan en el sitio web.
- Application: Almacena y comparte información para su uso durante una aplicación activa.

Cada objeto de ASP tiene propiedades y métodos que lo instruyen para la ejecución de una tarea específica. ASP a través de los objetos ADODB nos proporciona un acceso flexible a todo tipo de bases de datos, desde las sencillas, como Access hasta otras más potentes, como Oracle.

La librería ADODB se compone de tres objetos principales, Connection, RecordSet y Command, y de otros secundarios que son Field, Parameter, Record, Stream y Error. Con estos objetos se puede gestionar una base de datos mediante web de igual forma que se haría en una aplicación local.



### Ventajas de ASP

**Crear sitios web dinámicos.** Los sitios web dinámicos contienen páginas que muestran constantemente un contenido cambiante. Cuando se usa ASP se puede determinar el contenido que muestra una página web, dependiendo de muchos factores (la hora, contadores, fecha, ubicación, etc.) Las páginas web dinámicas son más útiles para cada usuario individual que las estáticas.

**Crear sitios web interactivos.** Estos incluyen páginas que intercambian información entre el sitio y el usuario. ASP les permite a los administradores de la web crear fácilmente páginas que procesan información de un usuario y luego generan contenido dependiendo de los datos suministrados por este. Los sitios web interactivos permiten a los administradores de la web personalizar el contenido de las páginas para atraer mejor al usuario.

**Crear aplicaciones.** Muchas personas están acostumbradas a trabajar con aplicaciones, como procesadores de palabras, agendas electrónicas y lectores de correo electrónico en sus computadoras. Con la tecnología ASP es posible crear aplicaciones que son albergadas en un sitio web y son accedidas por un explorador. Esto les permite a los usuarios correr programas escritos en lenguajes que no son soportados por sus exploradores de la web y acceder a una gama más amplia de programas.

**Usar componentes incorporados.** ASP incluye componentes fáciles de usar que pueden emplearse para realizar tareas tales como avisos rotativos, rastrear estadísticas de usuarios o verificar las características del explorador de la web de un usuario para que una página pueda ser personalizada en este. ASP les permite a los administradores de los sitios web crear y usar sus propios componentes en las páginas de ASP que crean.



**Trabajar con bases de datos.** Una opción importante de ASP es su capacidad de conectarse a una base de datos. Las páginas de ASP pueden usarse para que su información almacenada en una base de datos este disponible para los usuarios, este es un método eficiente de mostrar información actualizada en un sitio web. ASP también permite a los usuarios manipular la información de una base de datos (borrar, editar, crear registros).

**Mayor seguridad.** Debido a que el código de ASP es procesado en un servidor web, el usuario no puede acceder al código usado para crear las páginas de ASP. Esto le permite trabajar de una forma más segura con datos sensitivos, como nombres de registros y contraseñas. Si un usuario visualiza el código fuente de una página ASP dentro de un explorador, todo lo que verá es el código de html que fue generado por el servidor web para crear la página y no el código de ASP en sí.

### **Desventajas de ASP**

Por la forma en que esta estructurada la función de ASP, es necesario tener una línea bien definida y organizada desde el principio de la programación así no se tienen que enfrentar problemas al momento de tener que incorporar nuevos módulos en un futuro y se puede facilitar el manejo de varios administradores.

Por ser una tecnología creada principalmente para plataformas basadas en Windows se tienen que incorporar otras aplicaciones para que pueda ser utilizado en otras plataformas como los son Mac, Solaris, HP y se transforma en un costo adicional.

Su uso mediante el bloc de notas es muy rudimentario y para agilizar la programación de las páginas ASP es necesario contar con herramientas adicionales como el DreamWeaver, lo cual implica gastos adicionales.





En general, son más las ventajas que ofrece la utilización de ASP para el diseño de páginas web dinámicas, es por este motivo que se ha elegido para el sistema que se va a desarrollar.

### **2.4 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE SQL SERVER 2000**

SQL Server es un sistema administrador para bases de datos relacionales (RDBMS) basadas en la arquitectura cliente/servidor que usa Transact-SQL para mandar peticiones entre un cliente y el SQL Server.

#### **Arquitectura cliente/servidor**

SQL Server usa la arquitectura cliente/servidor para separar la carga de trabajo en tareas que corran en computadoras tipo servidor y tareas que corran en computadoras tipo cliente:

- El cliente es responsable de la parte lógica y de presentar la información al usuario. Generalmente, el cliente corre en una o más computadoras cliente, aunque también puede correr en una computadora servidor con SQL Server.
- SQL Server administra bases de datos y distribuye los recursos disponibles del servidor (tales como memoria, operaciones de disco, etc.) entre las múltiples peticiones.

#### **Sistema administrador para bases de datos relacionales (RDBMS):**

El RDBMS es responsable de:

- Mantener las relaciones entre la información y la base de datos.



- Asegurarse de que la información es almacenada correctamente, es decir, que las reglas que definen las relaciones entre los datos no sean violadas.
- Recuperar toda la información en un punto conocido en caso de que el sistema falle.

### **Transact – SQL**

Es una versión de SQL (Structured Query Language) usado como lenguaje de programación para SQL Server. SQL es un conjunto de comandos que permite especificar la información que se desea restaurar o modificar. Con Transact – SQL se puede tener acceso a la información, realizar búsquedas, actualizar y administrar sistemas de bases de datos relacionales.

### **Plataformas y requisitos para SQL Server 2000**

- Sistema operativo: Microsoft Windows NT® Server versión 4.0 Service Pack 5 (SP5) o posterior, Microsoft Windows NT Server 4.0 Enterprise Edition con SP5 o posterior, Microsoft Windows® 2000 Server, Microsoft Windows 2000 Advanced Server y Microsoft Windows 2000 Datacenter Server.
- Memoria: 64 MB de memoria RAM; se recomienda 128 MB.
- Disco duro: 95-270 MB de espacio libre para el servidor; 250 MB para la instalación típica
- Clientes compatibles: Windows 95, Windows 98, Windows Me, Windows NT Workstation 4.0 y Windows 2000 Professional. UNIX, Apple Macintosh y OS/2 requieren software de cliente de conectividad abierta de base de datos (ODBC) de otro fabricante.



### Limitantes de SQL Server 2000

En la Tabla 2.4.1 se describen las limitantes conocidas de SQL Server.

Característica	SQL Server 2000 (bytes)
column name length	128
index name length	128
table name length	128
max indexes per table	250
index length	900
max index column length	900
columns per index	16
max char() size	8000
max varchar() size	8000
max blob size	2147483647
max number of columns in GROUP BY	Limited only by number of bytes (8060)
max number of columns in ORDER BY	Limited only by number of bytes (8060)
tables per SELECT statement	256
max columns per table	1024
max table row length	8036
longest SQL statement	16777216
constant string size in SELECT	16777207

Tabla 2.4.1 Limitantes de SQL Server

### Servicios de SQL Server

**MSSQL Server.** Este servicio es el motor de la base de datos. Este es el componente que procesa todas las declaraciones de Transact-SQL y administra



todos los archivos que definen a la base de datos dentro del Servidor. Sus características son:

- Asignar los recursos de la computadora a múltiples usuarios simultáneos.
- Previene problemas lógicos, tales como sincronización de peticiones de usuarios que desean actualizar la misma información al mismo tiempo.
- Garantiza la integridad y consistencia de datos.

**SQLServer agent.** Este es un servicio que trabaja conjuntamente con SQL Server para crear y administrar las siguientes tareas:

- **Administración de alertas.** Las alertas brindan información acerca del estado de un proceso, tal como cuando un trabajo está completo o cuando ocurre un error.
- **Notificación.** El agente de SQL Server puede enviar e-mail, o iniciar otra aplicación cuando ocurre una alerta, por ejemplo, se puede programar una alerta para que ocurra cuando una base de datos o cuando una transacción está casi completa o cuando un respaldo de la base de datos ha terminado exitosamente.
- **Ejecución de tareas.** El agente de SQL Server incluye un motor de creación y planeación de tareas. Las tareas pueden ser simples operaciones de un solo paso, o pueden ser tareas complejas de varios pasos que requieren planeación. También se pueden crear pasos de las tareas con Transact-SQL, lenguajes script, o comandos del sistema operativo.
- **Administración de réplicas.** La replicación es el proceso de copiar datos o transacciones de un servidor a otro. El agente de SQL Server es el responsable de sincronizar los datos entre los servidores, monitorear los datos para buscar cambios y replicar la información en otros servidores.



**Microsoft distributed transaction coordinator.** Coordina la correcta realización de las transacciones distribuidas para asegurar que todas las actualizaciones en todos los servidores son permanentes; o en caso de errores, que las modificaciones sean canceladas.

**Microsoft search.** El soporte Full Text involucra la habilidad de emitir consultas hacia los datos y la creación y mantenimiento de índices que facilitan dichas consultas.

**Administración.** SQL Server provee una variedad de herramientas de administración para minimizar y automatizar las tareas administrativas rutinarias. Las declaraciones de Transact-SQL son el mecanismo interno usado para administrar SQL Server.

SQL Server puede ser administrado usando:

- Utilidades batch incluidas en SQL Server, tales como OSQL o BCP.
- Herramientas de administración gráfica incluidas en SQL Server.
- Aplicaciones COM-compatibles, tal como Visual Basic.

### **Seguridad en SQL Server**

SQL Server valida a los usuarios con 2 niveles de seguridad; autenticación del login y validación de permisos en la base de datos de cuentas de usuarios y de roles. La autenticación, identifica al usuario que está usando una cuenta y verifica sólo la habilidad de conectarse con SQL Server. El usuario debe tener permiso para acceder a las bases de datos en el Servidor. Esto se cumple para asignar permisos específicos para la base de datos, para las cuentas de usuario y los roles. Los permisos controlan las actividades que el usuario tiene permitido realizar en la base de datos de SQL Server.

---



**Modo de autenticación.** Cuando SQL Server está corriendo en Windows 2000 o NT, un sistema administrador puede especificar que está corriendo en uno de 2 modos de autenticación:

- **Modo de autenticación de Windows 2000 o NT.** Sólo está autorizada la autenticación de Windows NT. Los usuarios no pueden usar cuentas de SQL Server.
- **Modo mixto.** Cuando se usa este modo de autenticación, los usuarios se pueden conectar a SQL Server con la autenticación de Windows 2000 / NT o con la de SQL Server.

### **Cuentas de usuario y roles en una base de datos**

Después de que los usuarios han sido autenticados, y se les ha permitido conectarse a SQL Server, deben tener cuentas en la base de datos. Las cuentas de usuario y los roles, identifican permisos para ejecutar tareas.

**Cuentas de usuarios de las bases de datos.** Las cuentas de usuario utilizadas para aplicar permisos de seguridad son las de usuarios, o grupos de Windows o las de SQL Server. Las cuentas de usuario son específicas para cada base de datos.

**Roles.** Permiten reunir a los usuarios en una sola unidad a la cual se le pueden aplicar permisos. SQL Server contiene roles de servidor y de base de datos predefinidos para tareas administrativas comunes, de manera que pueden asignársele determinados permisos administrativos a un usuario en particular. También se pueden crear roles de base de datos definidos por el usuario. En SQL Server, los usuarios pueden pertenecer a varios roles:



- Roles fijos del Servidor. Proveen agrupamientos con privilegios administrativos a nivel del Servidor. Son administrados independientemente de las bases de datos de usuarios a nivel servidor
- Roles fijos de la base de datos. Proveen agrupamientos con privilegios administrativos a nivel de base de datos
- Roles de usuarios definidos en la base de datos. También se pueden crear roles para base de datos, para representar un trabajo desarrollado por un grupo de empleados dentro de una organización. No es necesario asignar y quitar permisos a cada persona. En función de que cambia un rol, se pueden cambiar fácilmente los permisos del rol y hacer que los cambios se apliquen automáticamente a todos los miembros del rol

### **Validación de permisos**

SQL Server realiza los siguientes pasos en la validación:

- Cuando el usuario realiza una acción, tal como ejecutar un comando de Transact-SQL o elegir una opción de un menú, los comandos de Transact SQL son enviadas al SQL Server.
- Cuando SQL Server recibe un comando de Transact –SQL, checa que el usuario tenga permisos para ejecutar dicha instrucción.
- Después, SQL realiza cualquiera de las siguientes acciones:
  - Si el usuario no tiene los permisos adecuados, SQL Server devuelve un error.
  - Si el usuario tiene los permisos adecuados, SQL Server realiza la acción.



### Bases de datos en SQL Server

Cada SQL Server tiene dos tipos de bases de datos: bases de datos del sistema y bases de datos del usuario. Las bases de datos del sistema almacenan información acerca de SQL Server como un total. SQL Server usa la base de datos del sistema para operar y administrar al sistema. Las bases de datos de usuarios, son bases de datos creadas por los usuarios. Una copia del SQL Server puede administrar una o más bases de datos de usuario. Cuando SQL Server es instalado, el instalador crea 4 bases de datos 2 de sistema y 2 de usuario (de ejemplo). La base de datos de distribución es instalada cuando se configura SQL Server para actividades de replicación.

### Ventajas y desventajas de SQL Server 2000

#### Ventajas

- Asignación Dinámica de Recursos. Permite la escalabilidad del uso del disco y memoria para adecuarse a las necesidades de la base de datos en cada momento. Esta flexibilidad permite un mejor rendimiento y simplifica la administración del software. La eliminación de dispositivos también es una ventaja añadida.
- Soporte 9x para Windows. El soporte para la plataforma Win9x aumenta significativamente la base de aplicaciones posibles para SQL Server.
- Analizador Gráfico de Consultas. Permite construir procedimientos almacenados y ejecutar consultas interactivas, también se pueden mostrar gráficamente los pasos que el procesador de consultas usa para ejecutar las tareas.
- Servicios OLAP del Servidor SQL de Microsoft. Permite el desarrollo de Data Warehouse, Data Mart, Data Mining entre otros.





- Servicios de Transformación de Datos (DTS). Simplifican la importación y la exportación de datos entre dos bases de datos compatibles con OLE DB. DTS también genera scripts Visual Basic (VBScript) que se pueden ejecutar desde el **WSH (Windows Scripting Host)** u otros entornos **COM (Component Object Model)**.
- Consolidar un pequeño número de servidores SQL liberará los recursos de hardware para otras aplicaciones, sin originar pérdida en las tareas de SQL Server.
- Soporte y actualizaciones por parte del proveedor.
- Generación de xml y compatibilidad con web services
- Compatible con .NET
- Escalabilidad a mediano y largo plazo.
- Totalmente habilitada para web
- Minería de datos (Data Mining)

### **Desventajas**

La instalación y operación requiere de sistemas operativos Windows, Internet Explorer y el licenciamiento para cada uno de los procesadores.

- Mejoras de seguridad mediante service packs que se basan en el sistema operativo.
- La migración requiere un reinicio de la base de datos. El reinicio de todos los datos en una base de datos, es un trabajo serio que invita a la potencial pérdida de datos. Cuanto más grande sea la base de datos, más onerosa será esta obligación.



- Si el crecimiento en las bases de datos de SQL Server tienden a tener un crecimiento exponencial, es necesario adquirir mas clusters que requieren mas procesador, disco y RAM

### **2.5 CARACTERÍSTICAS, VENTAJAS Y DESVENTAJAS DE LA COMUNICACIÓN VÍA WEB**

La comunicación web se basa en la transmisión de información en redes de computadoras, por lo que es importante conocer las topologías de redes existentes en la actualidad, entendiendo por topología de redes, el tipo de configuración que tienen.

Esta configuración recoge tres campos: físico, eléctrico y lógico. El nivel físico y eléctrico se puede entender como la configuración del cableado entre máquinas o dispositivos de control o conmutación. Cuando hablamos de la configuración lógica debemos pensar en como se trata la información dentro de nuestra red, como se dirige de un sitio a otro o como la recoge cada equipo.

#### **Topologías de redes**

##### **Topología en estrella**

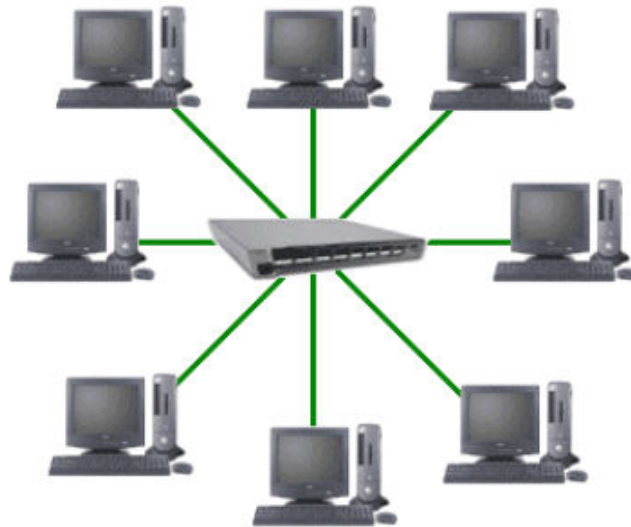
Todos los elementos de la red se encuentran conectados directamente mediante un enlace punto a punto al nodo central de la red, quien se encarga de gestionar las transmisiones de información por toda la estrella. Todas las tramas de información que circulen por la red deben pasar por el nodo principal, con lo cual un fallo en él provoca la caída de todo el sistema.



Por otra parte, un fallo en un determinado cable sólo afecta al nodo asociado a él; si bien esta topología obliga a disponer de un cable propio para cada terminal adicional de la red. La topología de Estrella es una buena elección siempre que se tenga varias unidades dependientes de un procesador, esta es la situación de una típica mainframe, donde el personal requiere acceder frecuentemente a esta computadora.

En este caso, todos los cables están conectados hacia un solo sitio, esto es, un panel central. Se puede cambiar, quitar o agregar algún equipo, sin que la red deje de funcionar.

Equipos como unidades de multiplexaje, concentradores y pares de cables sólo reducen los requerimientos de cableado, sin eliminarlos y producen alguna economía para esta topología. Resulta económica la instalación de un nodo cuando se tiene bien planeado su establecimiento, ya que este requiere de un cable desde el panel central, hasta el lugar donde se desea instalarlo. (Ver Figura 2.5.1).



**Figura 2.5.1. Topología en Estrella**



### **Topología de bus**

En esta topología, los elementos que constituyen la red se disponen linealmente, es decir, en serie y conectados por medio de un cable: el bus. Las tramas de información emitidas por un nodo (terminal o servidor) se propagan por todo el bus (en ambas direcciones), alcanzando a todos los demás nodos. Cada nodo de la red se debe encargar de reconocer la información que recorre el bus, para así determinar cual es la que le corresponde, la destinada a él.

Es el tipo de instalación más sencillo y un fallo en un nodo no provoca la caída del sistema de la red. Por otra parte, una ruptura del bus es difícil de localizar (dependiendo de la longitud del cable y el número de terminales conectadas a él) y provoca la inutilidad de todo el sistema.

Al transmitir una computadora, la señal se difunde a través del cable en ambos sentidos; se utilizan terminadores para el cable en los extremos, para que se disipe la señal y con esto evitar que siga circulando, de no existir este terminador en alguno de los extremos, la señal rebota produciendo una señal fantasma que altera la señal original, deformando su información, ocasionando que la red no funcione.

Como ejemplo más conocido de esta topología, encontramos la red Ethernet de Xerox. El método de acceso utilizado es el CSMA/CD, método que gestiona el acceso al bus por parte de los terminales y que por medio de un algoritmo resuelve los conflictos causados en las colisiones de información.

Cuando un nodo desea iniciar una transmisión, debe en primer lugar escuchar el medio para saber si está ocupado, debiendo esperar en caso afirmativo hasta que quede libre. Si se llega a producir una colisión, las estaciones reiniciarán cada una su transmisión, pero transcurrido un tiempo aleatorio distinto para cada estación.



Actualmente se encuentran implementadas cantidad de variantes de dicho método con sus respectivas peculiaridades.

El bus es la parte básica para la construcción de redes Ethernet y generalmente consiste de algunos segmentos de bus unidos ya sea por razones geográficas, administrativas u otras. (Ver Figura 2.5.2).



**Figura 2.5.2. Topología de Bus**

### **Topología en anillo**

Los nodos de la red se disponen en un anillo cerrado, conectados a él mediante enlaces punto a punto. La información describe una trayectoria circular en una única dirección y el nodo principal es quien gestiona conflictos entre nodos al evitar la colisión de tramas de información.

En este tipo de topología, un fallo en un nodo afecta a toda la red aunque actualmente hay tecnologías que permiten mediante unos conectores especiales, la desconexión del nodo averiado para que el sistema pueda seguir funcionando.

La topología de anillo esta diseñada como una arquitectura circular, con cada nodo conectado directamente a otros dos nodos. Toda la información de la red pasa a través de cada nodo hasta que es tomado por el nodo apropiado.



Este esquema de cableado muestra alguna economía respecto al de estrella. El anillo es fácilmente expandido para conectar más nodos, aunque en este proceso interrumpe la operación de la red mientras se instala el nuevo nodo. Así también, el movimiento físico de un nodo requiere de dos pasos separados: desconectar para remover el nodo y otra vez reinstalar el nodo en su nuevo lugar. (Ver Figura 2.5.3).



**Figura 2.5.3. Topología en Anillo**

### **Topología en estrella extendida**

La topología en estrella extendida es igual a la topología en estrella, con la diferencia de que cada nodo que se conecta con el nodo central también es el centro de otra estrella.

Generalmente el nodo central está ocupado por un hub o un switch, y los nodos secundarios por hubs. La ventaja de esto es que el cableado es más corto y limita la cantidad de dispositivos que se deben interconectar con cualquier nodo central.

La topología en estrella extendida es sumamente jerárquica, y busca que la información se mantenga local. Esta es la forma de conexión utilizada actualmente por el sistema telefónico.



### Topología en árbol

La topología en árbol es similar a la topología en estrella extendida, salvo que no tiene un nodo central. En cambio, utiliza un nodo de enlace troncal, generalmente ocupado por un hub o switch, desde el que se ramifican los demás nodos.

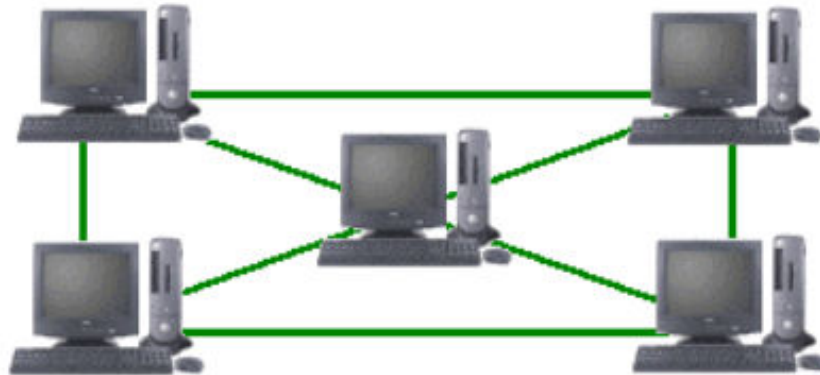
El enlace troncal es un cable con varias capas de ramificaciones, y el flujo de información es jerárquico. Conectado en el otro extremo al enlace troncal generalmente se encuentra un host servidor. (Ver Figura 2.5.4).



Figura 2.5.4. Topología de Árbol

### Topología en malla completa

En una topología de malla completa, cada nodo se enlaza directamente con los demás nodos. Las ventajas son que, como cada nodo se conecta físicamente a los demás, creando una conexión redundante, si algún enlace deja de funcionar la información puede circular a través de cualquier cantidad de enlaces hasta llegar a su destino. Además, esta topología permite que la información circule por varias rutas a través de la red. (Ver Figura 2.5.5).



**Figura 2.5.5. Topología de Malla Completa**

La desventaja física principal es que sólo funciona con una pequeña cantidad de nodos, ya que de lo contrario, la cantidad de medios necesarios para los enlaces y la cantidad de conexiones con los enlaces se torna abrumadora.

#### **Modelo TCP/IP**

La capa de interfase de red, comprende la capa física y de enlace de datos del modelo OSI (**Open Systems Interconnection**) se centra en el envío de datos hacia otros dispositivos. Se debe conocer la parte física de la red para formatear los datos correctamente y conocer sus desventajas y requerimientos.

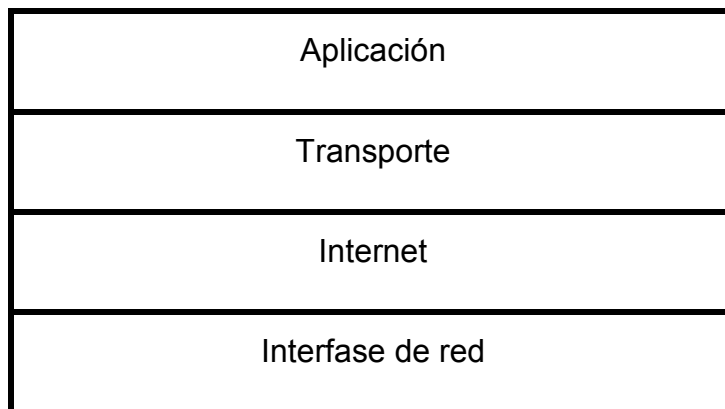
La capa de internet, abarca la capa de red del modelo OSI, proporcionando la funcionalidad para las comunicaciones entre redes a través de routers. Para realizar esto la capa de internet depende del protocolo IP, **protocolo de internet (Internet Protocol)**, el protocolo más importante de la suite TCP/IP.

La capa de transporte, esta capa es responsable de las comunicaciones extremo-extremo de la red, utiliza dos protocolos para realizar esta tarea: TCP, Protocolo de Control de Transmisión (Transmission Control Protocol) y UDP, Protocolo de Datagrama de Usuario (User Datagram Protocol).





La capa de aplicación, esta capa incluye todas las aplicaciones que hacen uso de la capa de transporte para enviar y recibir datos tales como RSH, Shell Remoto (Remote Shell); REXEC, Ejecución Remota (Remote Execute); TELNET, Emulación de Terminal (Terminal Emulation); FTP, Protocolo de Transferencia de Archivos (File Transfer Protocol), rlogin, DNS; NFS, Sistema de Archivos de Red (Network File System), etc. (Ver Figura 2.5.6).



**Figura 2.5.6 Modelo TCP/IP**

### **Direccionamiento lógico**

Se lleva a cabo en la capa de internet del TCP/IP, la cual es responsable de las funciones de conmutación y enrutamiento de la información (direccionamiento lógico), proporcionando los procedimientos necesarios para el intercambio de datos entre el origen y el destino, por lo que es necesario conocer la topología de la red (forma en que están interconectados los nodos), con objeto de determinar la ruta más adecuada.

Las principales funciones que se llevan a cabo, consisten en dividir los mensajes, conocer la topología de la red en caso de que las máquinas no se encuentren en la misma red, encaminar la información, enviar los paquetes y ensamblar los paquetes en el host destino.



### Encabezado IP

En la Figura 2.5.7 se muestra una representación del encabezado IP, separando los campos cada 32 bits.

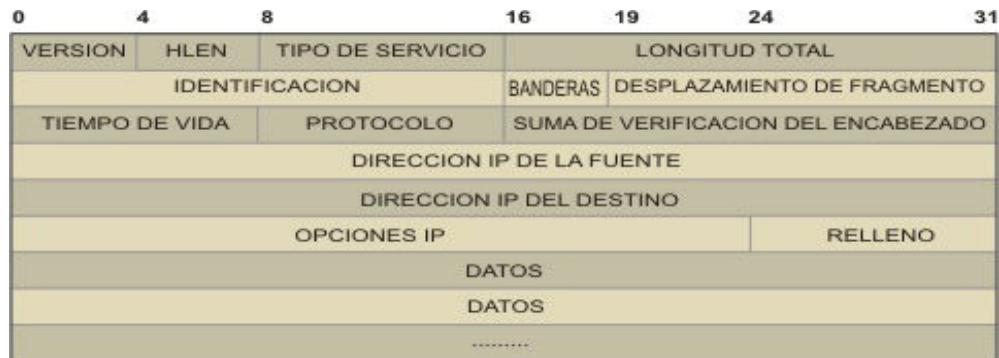


Figura 2.5.7 Encabezado IP

Los principales campos son:

- **Versión:** identifica la versión del protocolo IP que transporta el datagrama (4 en nuestro caso).
- **Tipo de Servicio:** 3 bits para la prioridad (0 al 7), bit 3 para RETARDOS CORTOS, bit 4 para ALTO DESEMPEÑO y el bit 5 para ALTA CONFIABILIDAD (los bits restantes permanecen sin uso).
- **Banderas:** 3 bits en total. El primero sin uso, el segundo es DF (DONT FRAGMENT), el tercero MORE FRAGMENTS (es decir que el datagrama en cuestión no es el último fragmento).
- **Tiempo de vida:** El número de saltos que podrá realizar el datagrama (un salto equivale al paso por un router). Originalmente se descontaban de este número tantas unidades como segundos le tomaran pasar por un router, actualmente se descuenta una unidad por router. (El valor máximo es 255).
- **Desplazamiento:** Distinto de cero cuando se trata de un fragmento (indica el desplazamiento respecto del primer fragmento).



- **Protocolo:** Contiene un valor que especifica el protocolo que lleva en la parte de datos del datagrama (UDP ó TCP por ejemplo).

### Encabezado TCP

Uno de los protocolos que puede viajar encapsulado por IP es TCP. En la Figura 2.5.8 vemos el encabezado TCP con el mismo formato que se utilizó para el encabezado IP. En este caso, los primeros campos son los de número de puerto de conexión, y los números de secuencia de los datos que salen y el número de acuse de recibo de los datos ya recibidos.

Un campo importante es el CODE BITS. En el que se activan o desactivan los flags del segmento TCP.

- **URG:** Flag urgente.
- **SYN:** Solicitud de sincronización.
- **RST:** Reset. Final de conexión.
- **ACK:** Acuse de recibo.
- **FIN:** El emisor ha llegado al final de su flujo de octetos.
- **PSH:** Empujar el paquete (priorizar).

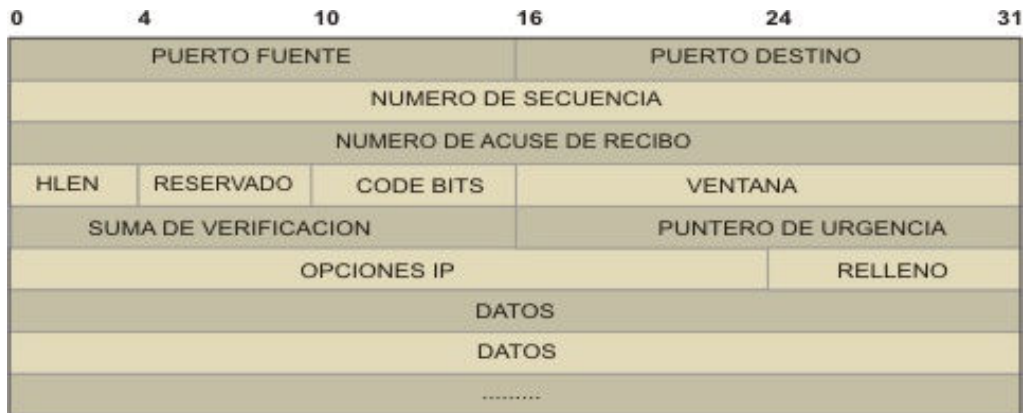


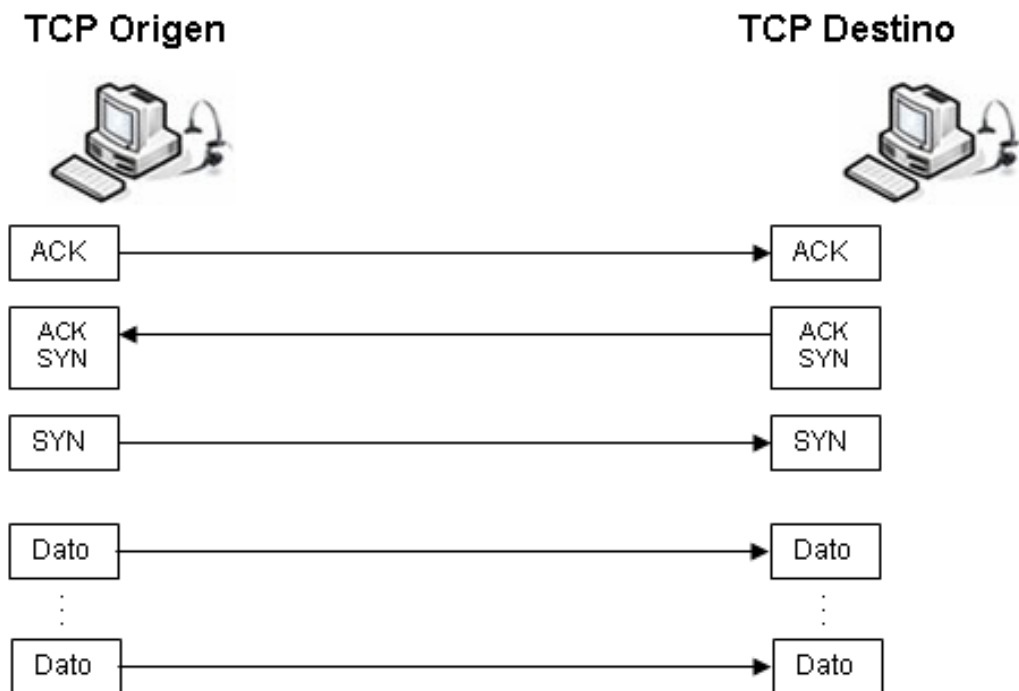
Figura 2.5.8 Encabezado TCP



Cuando un equipo quiere iniciar una sesión, TCP forma un paquete de sincronización (SYN), que incluye cosas como el tamaño del buffer, el número de secuencia que va a utilizar para los datos.

El TCP remoto (destino), al recibir un paquete SYN envía un paquete ACK-SYN y se lo envía al TCP origen.

El TCP origen, al recibir este paquete ACK-SYN, envía un paquete de ACK (confirmación) al equipo remoto, a partir de este momento la sesión está abierta. (Ver Figura 2.5.9).



**Figura 2.5.9. Transmisión de un mensaje entre de dos equipos a través de TCP/IP**



### Dispositivos de red

#### Hub o concentrador

Un concentrador es un dispositivo que permite centralizar el cableado de una red. También conocido con el nombre de hub. Un concentrador funciona repitiendo cada paquete de datos en cada uno de los puertos con los que cuenta, excepto en el que ha recibido el paquete, de forma que todos los puntos tienen acceso a los datos. También se encarga de enviar una señal de choque a todos los puertos si detecta una colisión. Son la base para las redes de topología tipo estrella. Como alternativa existen los sistemas en los que los ordenadores están conectados en serie, es decir, a una línea que une varios o todas las computadoras entre sí antes de llegar al equipo central. Llamado también repetidor multipuerto, existen 3 clases.

- **Pasivo:** No necesita energía eléctrica.
- **Activo:** Necesita alimentación.
- **Inteligente:** También llamados smart hubs son hubs activos que incluyen microprocesador.

El concentrador envía información a ordenadores que no están interesados. A este nivel sólo hay un destinatario de la información, pero para asegurarse de que la recibe el concentrador envía la información a todos los ordenadores que están conectados a él, así seguro que acierta.

Este tráfico añadido genera más probabilidades de colisión. Una colisión se produce cuando una computadora quiere enviar información y emite de forma simultánea con otro equipo que hace lo mismo. Al chocar los dos mensajes se pierden y es necesario retransmitir. Además, a medida que añadimos ordenadores a la red también aumentan las probabilidades de colisión. Un concentrador funciona a la velocidad del dispositivo más lento de la red; es un dispositivo simple,



esto influye en dos características. El precio es barato. Un concentrador casi no añade ningún retardo a los mensajes. (Ver Figura 2.5.10).



**Figura 2.5.10 Hub o concentrador**

### Switch

Un switch (en castellano "conmutador") es un dispositivo electrónico de interconexión de redes de computadoras que opera en la capa 2 (nivel de enlace de datos) del modelo **OSI**. Un conmutador interconecta dos o más segmentos de red, funcionando de manera similar a los puentes (bridges), pasando datos de un segmento a otro, de acuerdo con la dirección MAC de destino de los datagramas en la red.

Los conmutadores se utilizan cuando se desea conectar múltiples redes, fusionándolas en una sola. Al igual que los puentes, dado que funcionan como un filtro en la red, mejoran el rendimiento y la seguridad de las LANs (Local Area Network- Red de Área Local).

Los switches poseen la capacidad de aprender y almacenar las direcciones de red de nivel 2 (direcciones MAC) de los dispositivos alcanzables a través de cada uno de sus puertos. Por ejemplo, un equipo conectado directamente a un puerto de un switch provoca que el switch almacene su dirección MAC. Esto permite que, a diferencia de los concentradores o hubs, la información dirigida a un dispositivo se



dirija únicamente desde el puerto origen al puerto que permite alcanzar el dispositivo destino.

En el caso de conectar dos conmutadores o un switch y un concentrador, cada switch aprenderá las direcciones MAC de los dispositivos accesibles por sus puertos, por tanto en el puerto de interconexión se almacenan las MAC de los dispositivos del otro conmutador. (Ver Figura 2.5.11).



**Figura 2.5.11 Switch**

### **Gateway**

Una puerta de enlace o gateway es normalmente un equipo informático configurado para dotar a las máquinas de una red local (LAN) conectadas a él de un acceso hacia una red exterior, generalmente realizando para ello operaciones de traducción de direcciones IP (NAT: Network Address Translation).

Esta capacidad de traducción de direcciones permite aplicar una técnica llamada IP Masquerading (enmascaramiento de IP), usada muy a menudo para dar acceso a internet a los equipos de una red de área local compartiendo una única conexión a internet, y por tanto, una única dirección IP externa. (Ver figura 2.5.12).



**Figura 2.5.12 Gateway**

### **Bridge**

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

Un bridge conecta dos redes como una sola red usando el mismo protocolo de establecimiento de red. Funciona a través de una tabla de direcciones MAC detectadas en cada segmento a que está conectado. Cuando detecta que un nodo de uno de los segmentos está intentando transmitir datos a un nodo del otro, el bridge copia la trama para la otra subred. Por utilizar este mecanismo de aprendizaje automático, los bridges no necesitan configuración manual.

La principal diferencia entre un bridge y un hub es que por el segundo pasa cualquier trama con cualquier destino para todos los otros nodos conectados, en cambio por el primero sólo pasan las tramas pertenecientes a cada segmento. Esta característica mejora el rendimiento de las redes al disminuir el tráfico inútil. (Ver Figura 2.5.13).





**Figura 2.5.13** Brigde

### **Router**

El router o ruteador es un dispositivo de **hardware** para interconexión de redes de las computadoras que opera en la capa tres (nivel de red) del modelo OSI. El router interconecta segmentos de red, o algunas veces hasta redes enteras. Hace pasar paquetes de datos entre redes tomando como base la información de la capa de red.

El router toma decisiones basadas en diversos parámetros con respecto a la mejor ruta para el envío de datos a través de una red interconectada y luego dirige los paquetes hacia el segmento y el puerto de salida adecuados. Una de las más importantes es decidir la dirección de la red hacia la que va destinado el paquete (En el caso del protocolo IP esta sería la dirección IP).

Otras decisiones son la carga de tráfico de red en las distintas interfaces de red del router y establecer la velocidad de cada uno de ellos, dependiendo del protocolo que se utilice. (Ver Figura 2.5.14).



Figura 2.5.14 Router

La interacción de estos dispositivos en una red se muestra en la Figura 2.5.15.

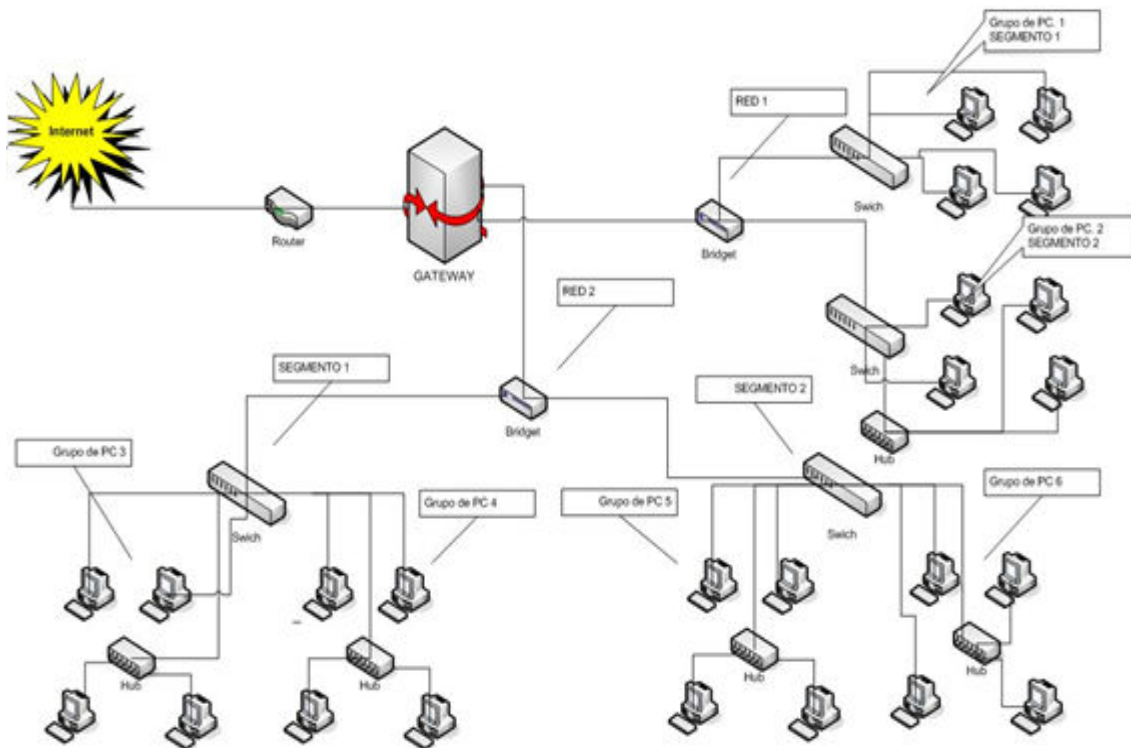


Figura 2.5.15 Interacción de los dispositivos en una red.

# **CAPÍTULO 3**

## **ANÁLISIS**



### 3.1 PROBLEMÁTICA ACTUAL

Para poder definir exactamente cuáles son los problemas que solucionará el sistema a desarrollar, se debe crear un diagrama que nos permita ver gráficamente cada uno de los pasos que debemos tener en cuenta, así como ubicar los puntos clave que nos ayuden a encontrar la mejor manera de resolverlos. (Ver Figura 3.1.1).

#### Diagrama de fases

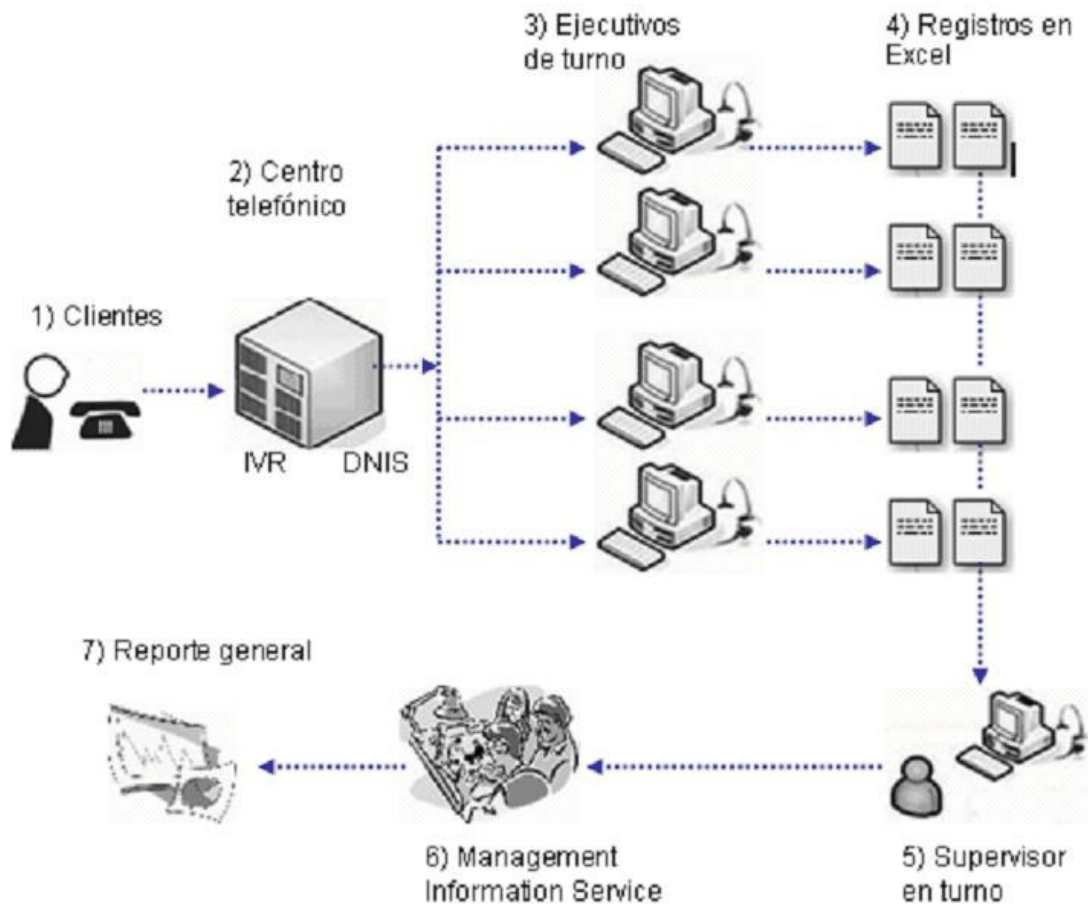


Figura 3.1.1 Fases de la problemática actual



### Definición de fases

- **Clientes.** Los clientes son las personas que han adquirido uno o varios productos de la marca, los cuales respondiendo a una campaña publicitaria llaman al número del centro telefónico de atención asignado, con su comprobante de compra y producto a la mano.
- **Centro telefónico.** Cuando la llamada entra es procesada técnicamente por un IVR que se encarga de filtrarla y de acuerdo a los últimos 4 dígitos del DNIS se enruta a un operador o ejecutivo disponible. A este proceso se le conoce como llamada inbound.
- **Operadores o ejecutivos.** En esta etapa los ejecutivos en turno reciben las llamadas de los clientes y en primer lugar identifican cual es el motivo de la llamada para canalizarla al registro de marca correspondiente como son:
  - ✓ Comentarios sobre productos
  - ✓ Servicios proporcionados por la marca
  - ✓ Promociones por temporada
  - ✓ Membresías a los clubes que maneja la marca en sus campañas
- **Registros.** Después que el ejecutivo identifica el tipo de servicio solicitado, procede a registrar los datos en hojas de Excel con los formatos que el supervisor asigno con anterioridad y sin importar que el cliente ya se haya comunicado anteriormente.
- **Supervisor en turno.** El proceso se sigue llevando a cabo durante todo el horario de atención del centro telefónico. Una vez concluido, cada ejecutivo vacía sus hojas de cálculo en una carpeta compartida ubicada en la PC del supervisor en turno. Finalizado el turno de la operación los datos son enviados por el supervisor al área de MIS.



- **Management Information Service (MIS).** En esta área es en donde se concentran todos los documentos generados por los ejecutivos para su depuración.
- **Reporte final.** Documento en Excel con toda la información obtenida y aun con algunos errores.

### **Problemática existente entre las fases planteadas**

#### **Incisos 1 y 2**

En esta etapa, los usuarios tienen que permanecer más tiempo en la línea de espera, porque técnicamente el **IVR (Interactive Voice Response System)** no ha detectado ejecutivos libres para canalizar la llamada, dando como resultado que el cliente se moleste y cuelgue debido a largos tiempos de espera desistiendo de participar en las promociones, generando la pérdida de información sobre esos productos.

#### **Incisos 2 y 3**

Cuando un ejecutivo queda libre, el IVR automáticamente le asigna una de las llamadas que están en espera, pero el proceso vuelve a detenerse ya que los ejecutivos tardan en registrar a los clientes por el proceso que se tiene de hojas de cálculo, esto puede llegar a saturar el IVR, generando el tono de ocupado en el servicio de atención.

#### **Incisos 3 y 4**

En esta etapa, los ejecutivos tienen que identificar el tipo de llamada que están recibiendo, ya que tienen que abrir un documento por cada una de las categorías posibles para proceder con el registro de los clientes en los formatos que el supervisor asignó antes de iniciar el turno, este proceso es lento ya que se tiene



que obtener información antes de empezar con los registros, identificando el documento.

El ejecutivo procede a obtener los siguientes datos los cuales son capturados en hojas de cálculo:

- Fecha del reporte
- Nombre
- Apellido paterno
- Apellido materno
- Número telefónico de casa y lada
- Número telefónico de oficina y lada
- Número telefónico móvil y lada
- Sexo
- Fecha de nacimiento
- Código postal
- Calle, número exterior e interior, colonia, delegación o municipio, estado
- Número de boleto/ticket
- Código
- Lote
- Fecha de caducidad
- Hora del reporte (verificada por el propio operador)
- Código adicional (si existe)
- Fecha de caducidad adicional
- Hora del reporte adicional (verificada por el propio operador)
- Monto



Primero se intenta buscar al cliente por medio de la función de búsqueda estándar de la hoja de cálculo, si no es encontrado, ya sea porque no existe o porque hubo un error al capturar sus datos, se inserta un nuevo registro.

Este proceso genera problemas tales como datos duplicados, ya que cada ejecutivo cuenta con su propia hoja de registro, errores de captura al no poderse verificar la existencia de código postal o colonias existentes, pérdida de tiempo en la búsqueda del cliente o en encontrar la hoja de registro según la promoción.

### **Incisos 4 y 5**

Cada uno de los ejecutivos genera un hoja de cálculo por cada categoría con cientos de registros, los cuales son depositados en la computadora del supervisor en turno, él se encarga de reunirlos todos tal y como fueron registrados, con errores y registros duplicados y los canaliza al área correspondiente para su procesamiento.

### **Incisos 5 y 6**

El supervisor entrega al área de MIS todos los documentos generados por los ejecutivos de servicio al cliente. Tomando en cuenta que existen entre 35 y 45 ejecutivos y que son 4 las modalidades para las cuales existe una hoja de cálculo, la cantidad de archivos entregados a dicha área oscila entre 136 y 180 con cientos de registros cada uno.

Todos los datos obtenidos son concentrados, revisados, depurados y corregidos. Se invierte gran cantidad de tiempo y todavía se cometen errores humanos en el manejo de la información.

### **Incisos 6 y 7**

Este trabajo, dará como resultado un archivo con toda la información recabada, que al contar con errores humanos podemos ver que no es 100% confiable y que

---





es muy difícil de depurar completamente. Con esta información se generan reportes diarios, semanales, mensuales o anuales que son enviados al área de mercadotecnia para su análisis.

### 3.2 REQUERIMIENTOS GENERALES Y ESPECÍFICOS

Los requerimientos del sistema son la pieza fundamental en un proyecto de desarrollo de software, y sus principales características son las siguientes:

- Planear el proyecto y los recursos que se usarán en él. Los líderes de proyecto utilizan los requerimientos como una base para la estimación del esfuerzo necesario en un proyecto.
- Se especifica el tipo de verificaciones que se habrán de realizar al sistema.
- Planear la estrategia de prueba a la que habrá de ser sometido el sistema. Los requerimientos son la base sobre la cual se decide si un caso de prueba fue ejecutado exitosamente por el sistema o no.
- Son el fundamento del ciclo de vida del proyecto. Los requerimientos documentados son la base para crear la documentación del sistema.

#### Características de un requerimiento

Una vez que se visualizo la importancia de los requerimientos en un sistema de software, entonces se deben definir las características que deben de poseer:

- Deben ser especificados por escrito. Bajo un acuerdo entre las dos partes.
- Deben ser posibles de probar o verificar. Si un requerimiento no se puede comprobar, entonces ¿cómo sabemos si cumplimos con él o no?.



- Descritos como una característica del sistema a entregar. Esto es: que es lo que el sistema debe de hacer (y no como debe de hacerlo).
- Lo más abstracto y conciso posible para evitar malas interpretaciones

### **Requerimientos generales**

Los requerimientos generales para el sistema son los siguientes:

- El sistema funcionará a través de cualquier navegador comercial de Internet.
- Para iniciar sesión se hará mediante un nombre de usuario y contraseña.
- Contará con una interfaz amigable.
- Se realizarán respaldos periódicos a la base de datos.
- La clave de cada usuario deberá tener un tiempo de expiración.
- Permitirá la generación de reportes para el análisis de la información.
- Permitirá un acceso rápido a la información del cliente.
- Enviará mensajes de término de sesión del usuario.
- Permitirá la validación de los campos para no introducir información errónea.
- Será una aplicación que corra bajo una plataforma Windows.
- Debe ser segura.
- Será escalable para permitir una actualización posterior.



### Requerimientos particulares

Los requerimientos particulares para el sistema son:

- Deberá permitir recopilar toda la información referente al cliente.
- Deberá permitir el ingreso de dos números telefónicos para localizar al cliente rápidamente.
- Permitirá validar la información capturada por los ejecutivos telefónicos.
- Deberá contener un catálogo que contenga los diferentes tipos de cierre de llamada.
- Permitirá el almacenamiento de la información de cada llamada no terminada.
- Deberá contener una pantalla para realizar búsquedas de clientes y evitar la duplicidad de los datos.
- Deberá mostrar en una pantalla toda la información relacionada con el cliente
- Enviará alertas cuando un campo haya sido llenado con información que no corresponda.
- Permitirá el almacenamiento del correo electrónico del cliente.
- Desplegará una alerta indicando el fin del proceso de captura de la información
- Al terminar de generar los reportes, se enviarán vía correo electrónico hacia el área de mercadotecnia.



### 3.3 ANÁLISIS DE LA INFORMACIÓN RECABADA

En el primer inciso de este capítulo se menciona la problemática actual que se presenta en el call center al realizar el levantamiento de información, recopilación de datos, concentración de la información y generación de reportes para su distribución y análisis. Los procesos anteriores son realizados de forma manual. El levantamiento de información es realizado a través de archivos de hojas de cálculo.

Cada ejecutivo del call center genera varios archivos de Excel, donde se capturan las llamadas que recibió durante el día, al finalizar su turno todos los archivos son enviados a los supervisores para que ellos los concentren y envíen al área de Administración de Servicios de Información (MIS). Dicha área se encarga de unir todos los archivos generados en el día y aplicarle los siguientes filtros: (Ver Figura 3.3.1).

	B	C	D	E	F	G	H	I
6	FECHA DE REGISTRO	No. DE CUENTE	NOMBRE	LADA	TELEFONO	CALLE	NEXT	N/INTERIOR
7	10/04/2004	1	MARIA DE LOURDES GONZALEZ ALVAREZ	55	55121955	SANTA GERTRUDIS	100	0
8	10/04/2004	2	MYRIAM ROMERO ROJAS	722	2272256	TERCERA PRIVADA DEL POZO	102	
9	10/04/2004	3	DANIEL AVILA FLORES	55	5309321	PROLONGACION 18 DE MARZO	349	
10	10/04/2004	4	JOSUE EMANUEL HERNANDEZ ZUÑIGA	55	57156092	GIRASOL	87	
11	10/04/2004	5	JOSUE EMANUEL HERNANDEZ ZUÑIGA	55	57156092	GIRASOL	87	
12	10/04/2004	6	JOSUE EMANUEL HERNANDEZ ZUÑIGA	55	57156092	GIRASOL	87	
13	10/04/2004	7	MARIA TERESA MONSERRAT CABALLERO RODRIGUEZ	722	2154025	BUENAVENTURA BERLIN	106	
14	10/04/2004	8	JOSUE EMANUEL HERNANDEZ ZUÑIGA	55	57156092	GIRASOL	87	
15	10/04/2004	9	WENDY LOPEZ RANGEL	55	59749536	MANZANA 3	ED. B	12
16	10/04/2004	10	MONICA PATRICIA MORENO CRUZ	413	36469783	AVENIDA CENTRAL	10	
17	10/04/2004	11	MONICA PATRICIA MORENO CRUZ	36469783	36469783	AVENIDA CENTRAL	10	
18	10/04/2004	12	MONICA PATRICIA MORENO CRUZ	36469783	36469783	AVENIDA CENTRAL	10	
19	10/04/2004	13	MONICA PATRICIA MORENO CRUZ	36469783	36469783	AVENIDA CENTRAL	10	
20	10/04/2004	14	KAREN ANGEL LONGORIA	962	6424069	CALANDRIA	5	MZ 6
21	10/04/2004	15	MARIA ELENA BERNAL ALVAER	722	2152045	FRANCISCO JAVIER MINA	116	
22	10/04/2004	16	OLIVIA MORFLOS ALBERTO	722	2109225	MANUEL MÉR Y TERAN	317	
23	10/04/2004	17	JUAN CARLOS DE LA CRUZ REYES	961	6045890	AV 13 DE SEPTIEMBRE MZ 9 LT 1	581	
24	10/04/2004	18	JAZMIN CHIMAL MEJIA	722	2180026	PASEO PRESIDENTE ADOLFO LOPEZ	311	
25	10/04/2004	19	JAZMIN CHIMAL MEJIA	722	2180026	PASEO PRESIDENTE ADOLFO LOPEZ	311	
26	10/04/2004	20	JAVIER GÓMEZ SOBERANIS	55	55419229	CERRO SANTA ISABEL	145	
27	10/04/2004	21	KRISTEN INGER SORSLAND LOPEZ	993	3541550	FRACCIONAMIENTO PALMITAS	6	DFT0401
28	10/04/2004	22	KRISTEN INGER SORSLAND LOPEZ	993	3541550	FRACCIONAMIENTO PALMITAS	6	DFT0401
29	10/04/2004	23	MARIA DE LOURDES OLIVARES GARCIA	55	56199692	MACARIO	562	6
30	10/04/2004	24	MARIA DE LOURDES OLIVARES GARCIA	55	56199692	MACARIO	562	6
31	10/04/2004	25	MARIA TERESA MONSERRAT CABALLERO RODRIGUEZ	722	2154025	BUENAVENTURA BERLIN	106	
32	10/04/2004	26	MARIA TERESA MONSERRAT CABALLERO RODRIGUEZ	722	2154025	BUENAVENTURA BERLIN	106	
33	10/04/2004	27	ELIZABETH BERENICE ALEJANDRE ROSALES	55	53573225	SAN FRANCISCO DE ASSIS	29	0
34	10/04/2004	28	SUSANA ARELLANO ALMANZA	422	2019161	TELEGRAFO	5	0
35	10/04/2004	29	SUSANA ARELLANO ALMANZA	422	2019161	TELEGRAFO	5	0

Eliminar registros duplicados

Figura 3.3.1 Hoja de cálculo que presenta información duplicada



- Ordenar la información por nombre del cliente
- Eliminar registros duplicados
- Identificar y eliminar registros con información inconsistente o incompleta

Una vez concentrada toda la información, se generan los reportes diarios, semanales, mensuales o anuales que serán enviados al área de mercadotecnia para un análisis posterior.

A continuación, con la información recabada se definirán las tablas que se utilizarán: se deberá crear una tabla para guardar el catálogo de los agentes que podrán acceder al sistema y deberá contener los campos: No. de agente, tipo (S-Supervisor, O-Operador), nombre, password. Y sólo podrá tener acceso el administrador del sistema, que es el usuario que se hará cargo de realizar las altas, bajas o cambios de los usuarios. (Ver tabla 3.3.1).

Nombre del campo	Tipo de dato	Longitud
Id_agente	int	4
Tipo	char	1
Nombre	varchar	50
Password	varchar	8

**Tabla 3.3.1 Descripción del catálogo de agentes**

Para almacenar la información del cliente, se deberá crear una tabla con los campos: identificador del cliente, nombre, apellido paterno, apellido materno, lada1, telefono1, lada2, telefono2, CP, calle, no. exterior, no. interior, colonia, delegación, ciudad, estado, fecha de nacimiento, correo electrónico. (Ver Tabla 3.3.2).



Nombre del campo	Tipo de dato	Longitud
Id cliente	int	4
Nombre	varchar	30
Apellido paterno	varchar	30
Apellido materno	varchar	30
Lada1	varchar	5
Tel1	varchar	10
Lada2	varchar	5
Tel2	varchar	10
CP	varchar	5
Calle	varchar	60
No ext	varchar	5
No int	varchar	5
Colonia	varchar	50
Delegacion	varchar	50
Ciudad	varchar	40
Estado	varchar	40
Fecha nacimiento	char	10
Email	varchar	60

**Tabla 3.3.2 Descripción de la tabla de clientes**

Una vez que se tiene capturada la información personal del cliente, el agente deberá preguntarle cierta información adicional como: si tiene hijos? , cuántos? si la respuesta fue afirmativa a la primera pregunta, el agente deberá preguntar si desea inscribirse al club? o preguntar si ya está inscrito?, cuál es su opinión del producto?, cuánto tiempo tiene de conocer el producto? y si tiene alguna sugerencia.

Dicha información deberá almacenarse en otra tabla junto con el identificador del agente que recibió la llamada y la fecha en la que se hizo el levantamiento de la información. (Ver Tabla 3.3.3).



Nombre del campo	Tipo de dato	Longitud
Tiene_hijos	char	2
Cuantos_hijos	int	4
Id_linea_producto	int	4
Opinión_producto	varchar	255
Tiempo_conocer_prod	varchar	10
Sugerencias	varchar	255
Inscrito_club	bit	1
Desea_insc_club	char	2
Id_agente	int	4
Fecha_registro	smalldatetime	4

**Tabla 3.3.3 Descripción de la tabla de datos adicionales**

Para que el levantamiento de la información sea más ágil, se crearán dos tablas más, una contendrá el catálogo de las líneas de productos que se estará manejando en la campaña con los campos: identificador de la línea de producto y su descripción (Ver Tabla 3.3.4) y la segunda, contendrá el catálogo con los posibles tipos de cierre de llamada, que contendrá los campos: identificador del cierre de llamada y descripción. (Ver Tabla 3.3.5).

Estos catálogos evitaran que el agente tenga que teclear esta información en cada llamada, ya que será desplegada a través de listas desplegables en los cuales podrá seleccionar la información deseada.

Nombre del campo	Tipo de dato	Longitud
Id_linea_producto	int	4
Descripción	varchar	50

**Tabla 3.3.4 Descripción del catálogo de productos**



Nombre del campo	Tipo de dato	Longitud
Id_cierre	int	4
Descripción	varchar	30

**Tabla 3.3.5 Descripción del catálogo de cierre de llamadas**

### 3.4 IDENTIFICACIÓN DEL PROBLEMA

En el planteamiento de la problemática existente del proceso actual del levantamiento de información de las campañas del call center, cuya función es recibir sugerencias, quejas y/o membresías a alguno de los clubes de la marca, surge la necesidad de contar con un sistema que permita recabar toda esta información, asimismo, el conocer las estadísticas emanadas de estos, para poder conocer los puntos donde se tienen fallas y corregirlas, o poner más atención en aquellos productos que reciben observaciones positivas y repetir esto en alguna otra línea de la marca. En el diagrama de fases de la Figura 3.1.1, vemos el flujo actual que se lleva a cabo en dicho proceso.

Los problemas generados se han identificado plenamente, y pueden ser solucionados; por esta razón, se plantea la automatización mediante el Sistema de Registro y Consulta de datos para solucionarlos. Estos problemas se pueden atacar de la siguiente forma (Ver Tabla 3.4.1):

Problemática	Solución
1. Los usuarios tienen que permanecer más tiempo en la línea de espera porque técnicamente el IVR no ha detectado ejecutivos libres para canalizar la llamada; dando	Automatizar el proceso de identificación del cliente mediante el almacenamiento de la información en una Base de Datos, funciones que permitan realizar búsquedas en el sistema por uno o más





<p>como resultado que el cliente se moleste y cuelgue debido a los largos tiempos de espera, desistiendo de participar en las promociones generando la perdida de información sobre los productos.</p>	<p>criterios, validación de la información que será almacenada, para conseguir la atención de una forma más rápida y, a través de esto, permitir la atención de un mayor número de clientes en menor tiempo.</p>
<p>2. Los ejecutivos tardan en registrar la información de los clientes por el proceso que se tiene de hojas de cálculo y esto puede llegar a saturar el Interactive Voice Response System (IVR) generando el tono de ocupado en el servicio de atención.</p>	<p>Creación de una pantalla que contenga campos de selección que brinden la oportunidad de seleccionar un dato en base a un catálogo, y con esto, evitar la captura manual de la información e incluso, elaborar pantallas que, en base a la información ingresada en un campo, permita el llenado automático.</p>
<p>3. Los ejecutivos tienen que identificar que tipo de llamada están recibiendo, ya que tienen que abrir un documento por cada una de las cuatro categorías posibles para proceder con el registro de los clientes en los formatos que el supervisor asigno antes de iniciar el turno, este proceso es lento, ya que se tiene que obtener información antes de empezar con los registros, identificando el documento; el ejecutivo procede a obtener los siguientes datos los cuales son capturados uno por uno en celdas de archivos de Excel. Este proceso genera problemas tales como</p>	<p>Identificación de los datos necesarios en base a selección del tipo de llamada, que puede ser una sugerencia o comentario, una suscripción a algún club de la marca o participar de alguna de las promociones que se tengan. En este caso, se podrá validar automáticamente la fecha de vigencia de la misma, por lo que se evitará el riesgo de aceptar la participación de algún cliente en promociones vencidas. Una vez identificado el tipo de llamada que se recibe, únicamente podrán ser ingresados datos relativos a la llamada, por medio de inhabilitación y/o</p>



<p>datos duplicados, debido a que cada ejecutivo cuenta con su propia hoja de registro, errores de captura al no poderse verificar la existencia de código postal o colonias existentes, errores en el reporte al ser verificado por el mismo ejecutivo, pérdida de tiempo en lo que se busca algún registro en las hojas de Excel.</p>	<p>ocultamiento de componentes del sistema. Mediante el ingreso del código postal se delimitará el número de colonias, y al seleccionar alguna de éstas, se delimitará el número de calles; en el caso de contar con el número de folio del cliente, en automático se cargarán los apellidos, nombre(s), dirección, entre otros.</p>
<p>4. Cada uno de los ejecutivos genera un documento de Excel por cada categoría con cientos de registros, los cuales son depositados en la computadora del supervisor en turno, quien se encarga de concentrarlos, tal y como fueron registrados con errores, registros duplicados, etc. y los canaliza al área correspondiente para su procesamiento.</p>	<p>La información será centralizada en la Base de Datos, permitiendo tener datos confiables, consistentes y oportunos, con ello, tener información útil y una buena administración de los recursos humanos.</p>
<p>5. El supervisor entrega al área de MIS, donde se procesa la información generada por los ejecutivos de servicio al cliente teniendo en cuenta que existen entre 35 y 45 ejecutivos, que son 4 las modalidades para las cuales existe un documento en Excel; la cantidad de documentos con cientos de registros oscilan entre 136 y 180</p>	<p>Mediante el almacenamiento de información en la Base de Datos, ya no habrá necesidad de depurar, corregir y completar los registros de los clientes, debido a que se establecerán validaciones que otorguen confiabilidad en la captura de los datos, además que tener dicha información organizada y disponible en base a los criterios y</p>



archivos, todos los datos obtenidos son depurados, corregidos y omitiendo registros incompletos invirtiendo gran cantidad de tiempo y cometiendo todavía errores humanos en el manejo de la información.	necesidades del call center, por lo que el ahorro de tiempo y de personal será significativo.
6. Este trabajo da como resultado un archivo con toda la información recabada que al contar con errores humanos podemos ver que no es 100% confiable y que es muy difícil de depurar completamente. Con esta información se generan reportes diarios, semanales, mensuales o anuales que son enviados al área de mercadotecnia para su análisis.	Elaboración de reportes de diversos criterios, que podemos mencionar el número de llamadas realizadas en el día, categoría en la que entran; reporte del número de quejas y tipo. Reporte mensual de llamadas captadas y el tiempo de duración de la elaboración de cada reporte, pudiendo incrementarse el número y tipo de los mismos, en base a las necesidades de la marca.

**Tabla 3.4.1 Soluciones planteadas a la problemática actual**

### 3.5. JUSTIFICACIÓN DE LA SOLUCIÓN PROPUESTA

La importancia del desarrollo del actual proyecto se divide en dos grupos que son el front-end y el back-end. En este diseño de software, el front-end es la parte del software que interactúa con el usuario y el back-end es la parte que procesa la entrada desde el front-end.

La separación del sistema en front ends y back ends es un tipo de abstracción que ayuda a mantener las diferentes partes del sistema separadas. La idea general es que el front-end es el responsable de recolectar los datos de entrada del usuario,



que pueden ser de muchas y variadas formas y procesarlas de una manera conforme a la especificación que el back-end pueda utilizar. La conexión del front-end y el back-end es un tipo de interfase.

### **Sistemas de gestión de bases de datos (back-end)**

Los sistemas de gestión de bases de datos son un tipo de software muy específico, dedicado a servir de interfase entre la base de datos, el usuario y las aplicaciones que la utilizan.

Se compone de un lenguaje de definición de datos, de un lenguaje de manipulación de datos y de un lenguaje de consulta. En los textos que tratan este tema, o temas relacionados, se mencionan los términos SGBD y DBMS, siendo ambos equivalentes y acrónimos, respectivamente, de Sistema Gestor de Bases de Datos y DataBase Management System, por sus siglas en inglés.

Razón por la cual se procede a la evaluación de los productos más representativos del mercado:

- MySQL
- Adaptive Server Enterprise (ASE- Sybase)
- SQL Server 2000

### **MySQL**

MySQL es un sistema de gestión de base de datos multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB desarrolla MySQL como software libre en un esquema de licenciamiento dual. Por un lado es gratuito, pero empresas que quieran incorporarlo en productos privados pueden comprar una licencia empresarial que les permita su uso.



MySQL es muy utilizado en aplicaciones web y es multiplataformas. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia.

En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones por sus características:

- Uso de multi-hilos mediante hilos del kernel
- Usa tablas en disco b-tree para búsquedas rápidas con compresión de índices
- Tablas hash en memoria ( temporales )
- Completo soporte para operadores y funciones en cláusulas select y where
- Completo soporte para cláusulas group by, order by y de funciones de agrupación
- Soporta gran cantidad de datos. MySQL Server tiene bases de datos de hasta 50 millones de registros.
- Se permiten hasta 64 índices por tabla. Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas
- Los clientes se conectan al servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows se pueden conectar usando named pipes y en sistemas Unix usando ficheros socket Unix
- MySQL contiene su propio paquete de pruebas de rendimiento proporcionado con el código fuente de la distribución de MySQL



### **Adaptive Server Enterprise**

Adaptive Server Enterprise (ASE) es el motor de bases de datos (RDBMS). Es un sistema de gestión de datos, altamente escalable, de alto rendimiento, con soporte a grandes volúmenes de datos, transacciones y usuarios, de bajo costo, que permite:

- Almacenar datos de manera segura
- Tener acceso y procesar datos de manera inteligente
- Movilizar datos.

La versión 15 de ASE incluye características nuevas como:

- Técnicas de particionamiento semántico de tablas que aumentan la velocidad de acceso a los datos
- Columnas cifradas para mayor seguridad de los datos
- Columnas computadas "virtuales" y materializadas, e índices funcionales, que brindan mayor rendimiento
- Mejoras al lenguaje Transact-SQL, para mayor productividad
- Mejoras a los servicios de Java y XML en la base de datos
- Mejoras a los servicios para consumo y publicación de Servicios web
- Herramientas mejoradas para la administración y el monitoreo
- Administrador lógico de recursos y tareas
- Múltiples esquemas de bloqueo de datos
- Copias de respaldo en línea y de alto rendimiento
- Integración transparente con orígenes de datos remotos



- Programador de tareas
- Conexiones seguras con SSL
- Soporte a múltiples herramientas de desarrollo y lenguajes de programación, como PowerBuilder, Visual Basic, Java, C, PHP, etc.

Entre sus desventajas podemos mencionar:

- Alto consumo de espacio en disco
- El precio de licenciamiento es muy caro en comparación con SQL Server o MySQL
- Su esquema de programación Sybase T-SQL no soporta vistas indexadas, manejo de múltiples triggers
- El tamaño definido para los nombres de base de datos, tablas, vistas, columnas, índices, stored procedures, entre otros es de 30 caracteres

### Comparativas de los manejadores de bases de datos

En la Tabla 3.5.1 se muestran las comparaciones de las diferentes herramientas hasta ahora presentadas:

Característica	SQL Server	MYSQL	ASE - Sybase
Robustez	Excelente	Excelente	Excelente
Multiplataforma	Regular	Excelente	Excelente
Disponibilidad de herramientas	Excelente	Bueno	Bueno
Integración de datos	Excelente	Bueno	Excelente
Compatibilidad con otros lenguajes de desarrollo	Bueno	Bueno	Bueno
Capacidad de almacenamiento	Bueno	Bueno	Bueno
Accesos múltiples	Excelente	Bueno	Excelente
Restauración de sistema	Excelente	Bueno	Bueno
Concurrencia	Excelente	Excelente	Excelente



Tiempo de importación y generación de índices	Excelente	Bueno	Bueno
Facilidad de uso	Excelente	Excelente	Bueno
Calidad de reportes	Excelente	Regular	Bueno
Compartir datos con otros DBMS	Excelente	Regular	Regular
Costo	Excelente	Excelente	Regular
Manejo de stored procedures	Excelente	Bueno	bueno

**Tabla 3.5.1 Comparación de los DBMS**

**Tecnologías de servidor (front-end)**

Este tipo de tecnologías utilizan software que ayuda al servidor web a procesar las páginas que contienen scripts o etiquetas del lado del servidor. Cuando se solicita al servidor una página de este tipo, el servidor web pasa la página al servidor de aplicaciones para su procesamiento antes de enviarla al navegador. Entre estas aplicaciones destacan:

- PHP
- JSP
- ASP

**PHP**

**PHP (Hypertext Preprocessor)** es un lenguaje script (no se compila si no que existe un intérprete que lee el código y se encarga de ejecutar las instrucciones que contiene éste código), para el desarrollo de páginas web dinámicas del lado del servidor, cuyos fragmentos de código se intercalan fácilmente en páginas html, debido a esto, y a que es **open source (código abierto)**, es el más popular y extendido en la web.

PHP es capaz de realizar determinadas acciones de una forma fácil y eficaz sin tener que generar programas en un lenguaje distinto a html. Esto se debe a que





PHP ofrece un extenso conjunto de funciones para la explotación de bases de datos sin complicaciones. Es por esto, que levanta un mayor interés con respecto a los lenguajes pensados para los CGI. Entre las ventajas podemos citar:

- Muy sencillo de aprender.
- Similar en sintaxis a C y a PERL
- Soporta en cierta medida la orientación a objeto. Clases y herencia.
- El análisis léxico para recoger las variables que se pasan en la dirección lo hace PHP de forma automática. Librándose el usuario de tener que separar las variables y sus valores
- Se puede incrustar código PHP con etiquetas html
- Excelente soporte de acceso a base de datos
- La comprobación de que los parámetros son válidos se hace en el servidor y no en el cliente (como se hace con javascript) de forma que se puede verificar que las solicitudes sean correctas. Además PHP viene equipado con un conjunto de funciones de seguridad que previenen la inserción de órdenes dentro de una solicitud de datos
- Se puede hacer de todo lo que se pueda transmitir por vía HTTP

Y las desventajas:

- Todo el trabajo lo realiza el servidor y no delega al cliente. Por tanto puede ser más ineficiente a medida que las solicitudes aumenten de número.
- La legibilidad del código puede verse afectada al mezclar sentencias html y PHP.
- La orientación a objetos es aún muy deficiente para aplicaciones grandes



### JSP

**Java Server Pages (JSP)**, o bien Páginas de Servidor Java. Es una tecnología orientada a crear páginas web con programación en Java. Con JSP se crean aplicaciones web que se ejecuten en variados servidores web de múltiples plataformas, ya que Java es en esencia un lenguaje multiplataforma. Las páginas JSP están compuestas de código html/xml mezclado con etiquetas especiales para programar scripts de servidor en sintaxis Java.

El motor de las páginas JSP está basado en los servlets de Java -programas en Java destinados a ejecutarse en el servidor-. En JSP se crean páginas de manera parecida a como se crean en ASP o PHP.

Se generan archivos con extensión .jsp que incluyen, dentro de la estructura de etiquetas html, las sentencias Java a ejecutar en el servidor. Antes de que sean funcionales los archivos, el motor JSP lleva a cabo una fase de traducción de esa página en un servlet, implementado en un archivo class (Byte codes de Java).

Entre las ventajas que éste posee, podemos citar:

- Utilización de procesos ligeros (hilos Java) para el manejo de las peticiones
- Manejo de múltiples peticiones sobre una página .jsp en un instante dado
- Facilidad para compartir recursos entre peticiones
- Creación, utilización y modificaciones de JavaBeans del servidor
- Cambios realizados en el código html relativos a cómo son mostrados los datos, no interfieren en la lógica de programación y viceversa
- Los diseñadores de páginas pueden centrarse en el código html y los programadores en la lógica del programa



Entre sus desventajas podemos citar dos de ellas:

- La excesiva complejidad deriva en enfocar cualquier problema como un EJB (enterprise java bean), lo que muchas veces lleva a consumir grandes cantidades de tiempo y dinero.
- La aproximación elemental radica en que muchas empresas usan JSP como ASP, es decir, como una manera de hacer sus páginas dinámicas pero sin integrarlas a todos los módulos Java

### Comparativas de las tecnologías de servidor

En la Tabla 3.5.2 se muestran las comparaciones entre las diferentes herramientas de programación web hasta ahora presentadas:

Característica	ASP	PHP	JSP
Desempeño	Excelente	Excelente	Excelente
Multiplataforma	Bueno	Excelente	Excelente
Disponibilidad de herramientas de	Excelente	Bueno	Bueno
Conectividad a diferentes bases de datos	Excelente	Bueno	Excelente
Documentación	Excelente	Excelente	Excelente
Soporte	Excelente	Excelente	Excelente
Depuración y control de errores	Bueno	Bueno	Excelente
Elaboración de pantallas	Excelente	Bueno	Bueno
Componentes ActiveX	Excelente	Bueno	Bueno
Facilidad de aprendizaje	Excelente	Excelente	Bueno
Costo	Excelente	Excelente	Excelente
Velocidad	Excelente	Excelente	Excelente

**Tabla 3.5.2 Comparación de las herramientas de programación web**



### **Elección de la herramienta para el desarrollo del back-end**

Dentro de las principales características que se han descrito anteriormente para cada una de las bases de datos, se ha optado por elegir a SQL Server 2000 y estas son algunas de las razones para la toma de la decisión:

- Bajo costo en comparación con SAE de Sybase
- Fácil instalación, uso y manejo
- Transact-SQL es un lenguaje de BD más poderoso en comparación con Sybase T-SQL, MySQL apenas ha estado iniciando en este desarrollo
- Generación de reportes mediante ReportServices
- Restauración del sistema
- Documentación disponible en la WWW

### **Elección de la herramienta para el desarrollo del front-end**

Para definir la elección de nuestra herramienta de desarrollo del front-end, se ha optado por elegir al ASP para la publicación del presente proyecto dentro del call center. Estas son algunas de las razones:

- Es gratuito al igual que las herramientas descritas anteriormente, siempre y cuando se cuente con las licencias de Windows.
- La infraestructura informática dentro del call center se basa en Windows
- Cuenta con la capacidad de trabajar con ActiveX
- La curva de aprendizaje es muy corta al utilizar VBScript



### **Justificación de la solución**

Al llegar a este punto se han definido nuestras herramientas tanto para el front-end cómo para el back-end, elegir uno de los manejadores de las bases de datos no es fácil, teniendo en cuenta la portabilidad del sistema, pero al manejar este esquema, se ha definido como parte primordial lo robusto de la base de datos, el costo en operación y su mantenimiento.

Por las características propias del lenguaje Transact-SQL el front-end puede en un futuro cambiar su entorno de desarrollo y migrar sin mayor complejidad a .NET o bien ser implementado en PHP. La elección de ASP se debe al nivel de conocimientos entre los desarrolladores, el bajo costo de capacitación y lo descrito anteriormente.

# **CAPÍTULO 4**

## **DISEÑO E IMPLEMENTACIÓN**



## 4.1 DISEÑO DEL SISTEMA

### 4.1.1 Diagrama de contexto

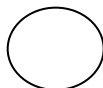
Un diagrama de contexto (DC) es un caso especial de un DFD (diagrama de flujo de datos), en donde un proceso único representa el sistema completo, define la frontera o marco del análisis o lo que es lo mismo, la interfaz del sistema y el resto del universo. Destaca varias características importantes del sistema:

- Aquellos agentes externos que se comunican con el sistema, que a veces se conocen como terminadores.
- Los datos que el sistema recibe del exterior y que se deben procesar de alguna forma.
- Los datos producidos por el sistema y que se mandan al mundo exterior.
- La frontera entre el sistema y el resto del mundo.
- Los almacenes de datos que el sistema comparte con los agentes externos. Estos almacenes se crean fuera del sistema para su uso, o bien son creados en él y usados fuera.

### Elementos de un diagrama de contexto

Los elementos de un diagrama de contexto son los mismos que los de un DFD:

**Sistema.** Representado por una burbuja. Indica una parte del sistema que transforma entradas en salidas. En este caso, representa el sistema entero. Trabaja sobre los datos. Debe tener un nombre claro, compuesto por un verbo de acción más un sustantivo.





**Entidad / agente externo.** Define la frontera del sistema. Son los emisores y/o receptores de los flujos de información y aparecen exclusivamente en el nivel 0 ó diagrama de contexto. No se puede cambiar el contenido de una entidad externa. Cualquier relación entre entidades externas no debe aparecer en el diagrama. Son la interfaz con el mundo exterior.



**Flujo de datos.** Representa los datos del sistema. Son flechas que indican el movimiento de los datos, definen origen, dirección y destino de los mismos. Unen al emisor y al receptor de los datos y conectan al resto de los componentes del DFD. Se pueden bifurcar o converger. Deben ser claros, concisos y fáciles de definir. Se debe evitar flujos diferentes con el mismo nombre.



A continuación en la Figura 4.1.1.1 tenemos el diagrama de contexto del sistema.

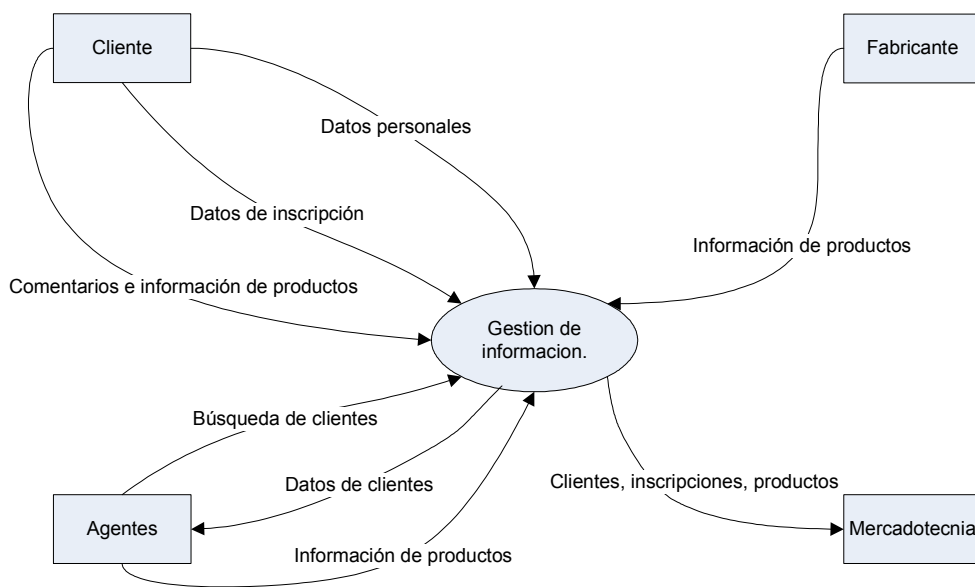


Figura 4.1.1.1 Diagrama de contexto del sistema





### 4.1.2 Diagrama de flujo

Es un modelo lógico-gráfico para representar el funcionamiento de un sistema. A continuación se presentan los diagramas correspondientes a los módulos del sistema. En la Figura 4.1.2.1 se muestra el diagrama de flujo general de la aplicación.

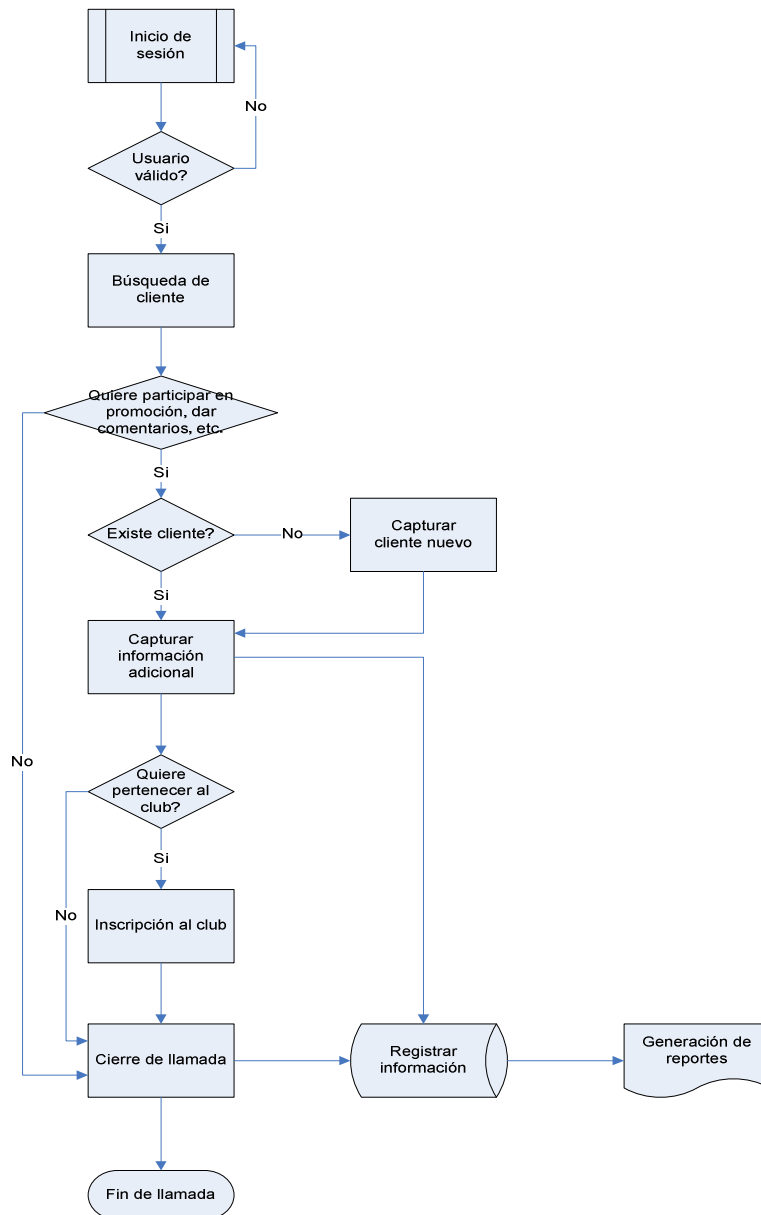


Figura 4.1.2.1 Diagrama de flujo general



Para mayor facilidad en la creación de nuestros algoritmos, hemos dividido el diagrama de flujo general en cada uno de los módulos que lo representan.

En la Figura 4.1.2.2 se tiene el diagrama para la administración de acceso al sistema, en él interactúan principalmente los agentes y, en segundo plano, el administrador que es quien definirá el acceso a la aplicación.

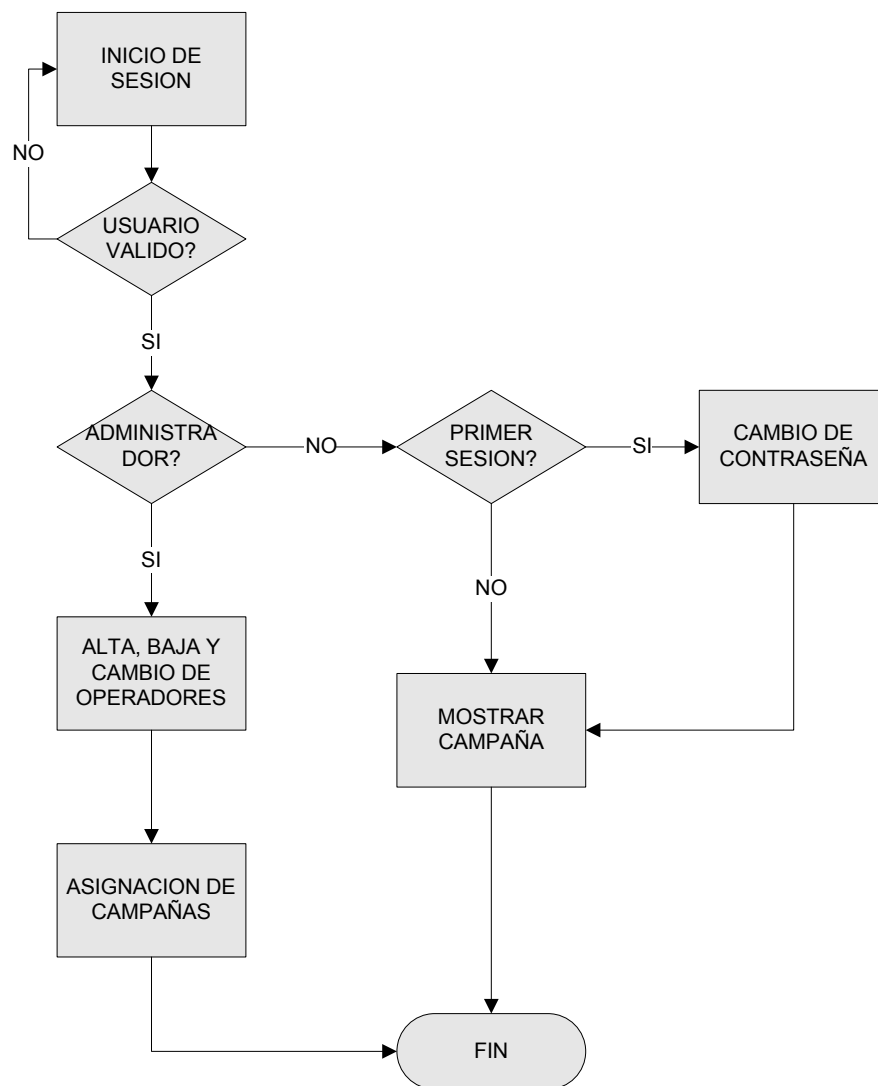


Figura 4.1.2.2 Diagrama de acceso al sistema



### 4.1.3 Diagrama de flujo de datos

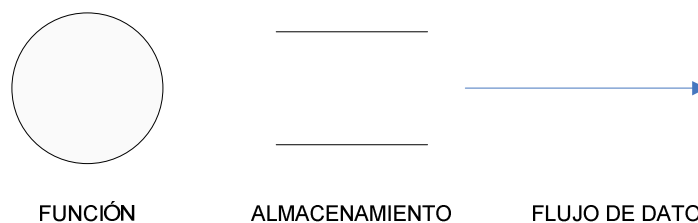
Un diagrama de flujo de datos (DFD) es una representación estructurada y gráfica que describe cómo circula la información a través de un sistema y los diferentes procesos de transformación a los que se ve sometida. Permite visualizar un sistema como una red de procesos funcionales conectados mediante flujos de datos.

Los elementos que componen a un DFD son los siguientes, representados en la Figura 4.1.3.1

**Función.** Representa la transformación del flujo de datos. Muestra como una o más entradas se transforman en salidas. Su nombre es lo suficientemente conciso y claro para que cualquier persona entienda de qué se trata. Dichas funciones van numeradas para diferenciarlas en un mismo nivel mostrando la jerarquía entre los niveles. Es representado por un círculo.

**Flujo de dato.** Es la representación del flujo del dato del origen al destino. Lo representa una flecha que indica su dirección. Los datos siempre van hacia y/o desde una función.

**Almacenamiento.** Son los datos pasivos; generalmente archivos, tablas, etc. Son representados por líneas paralelas ó cuadros.



**Figura 4.1.3.1 Elementos del diagrama de flujo de datos**

---



El DFD se basa en descomposiciones llamadas niveles. El primer nivel es una representación muy general del sistema. Aumentan los detalles en la medida en la que se alcanzan niveles más bajos (subniveles) en la descomposición.

En la Figura 4.1.3.2 tenemos el diagrama de flujo de datos general del sistema en su primer nivel y se descompondrá en el número de procesos incluidos en él.

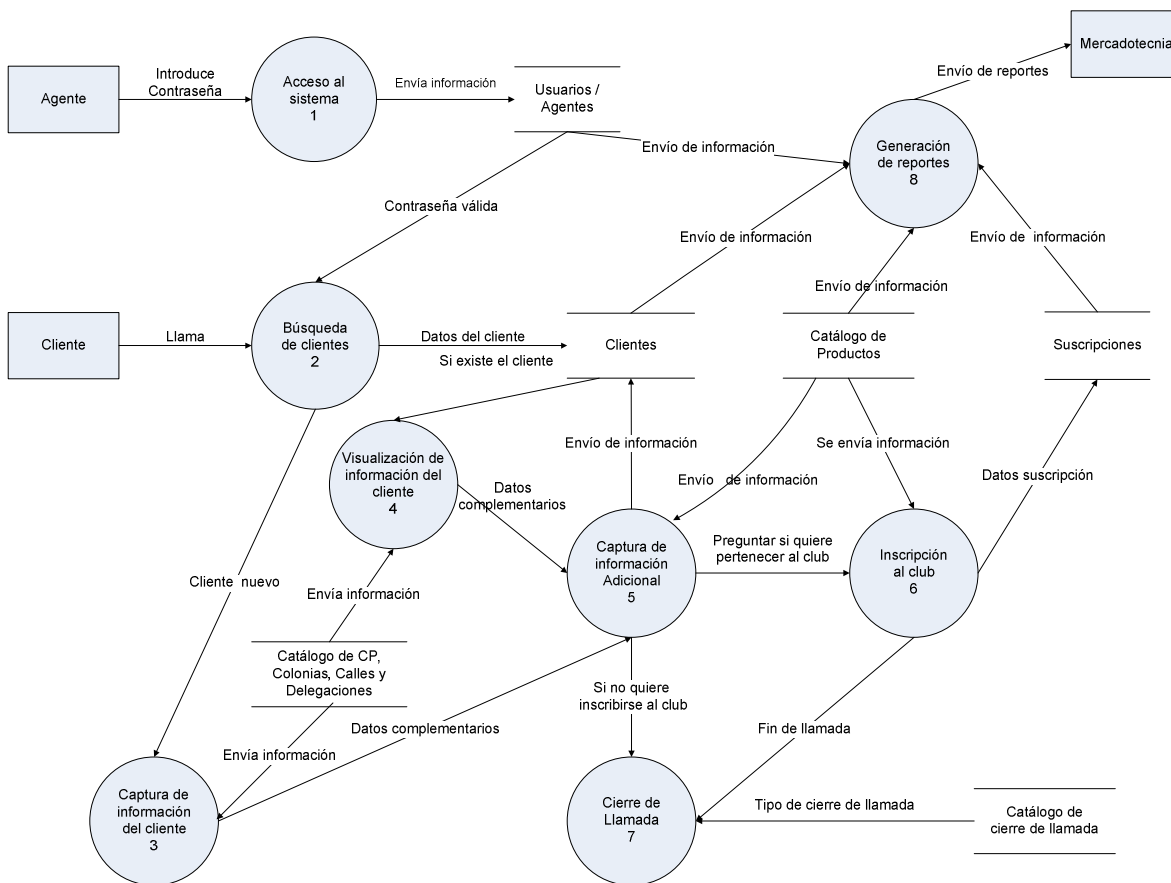


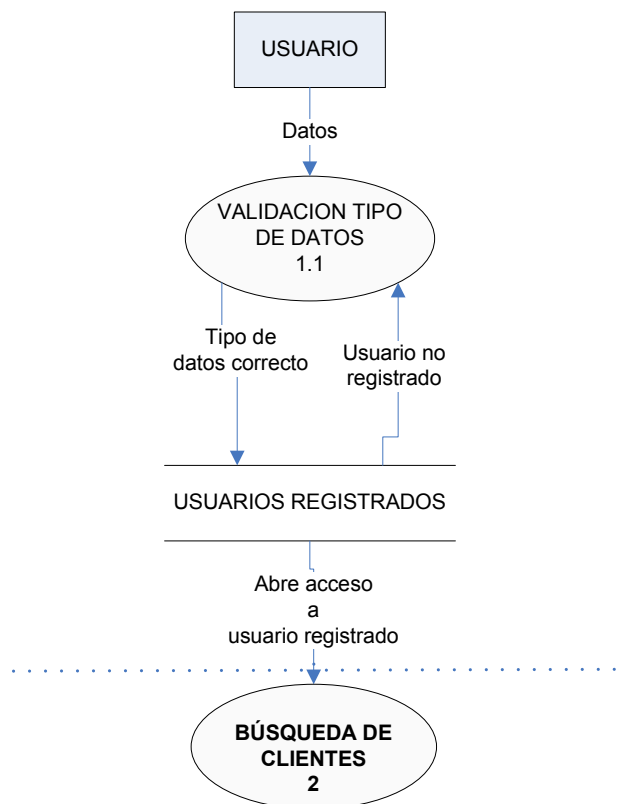
Figura 4.1.3.2 Diagrama de flujo de datos general del sistema

A continuación tenemos la descomposición del DFD por módulos y en cada uno de ellos se realizará su correspondiente descomposición.



### Acceso al sistema

En la Figura 4.1.3.3 tenemos el flujo de segundo nivel para el acceso al sistema.



**Figura 4.1.3.3 Flujo para la administración de accesos al sistema**

Cabe mencionar que el proceso de Validación tipo de datos tiene la tarea de validar que los datos a enviar por parte del usuario no contengan caracteres especiales que afecten el funcionamiento de la base de datos.



El DFD en segundo nivel para la búsqueda de clientes se describe en la Figura 4.1.3.4

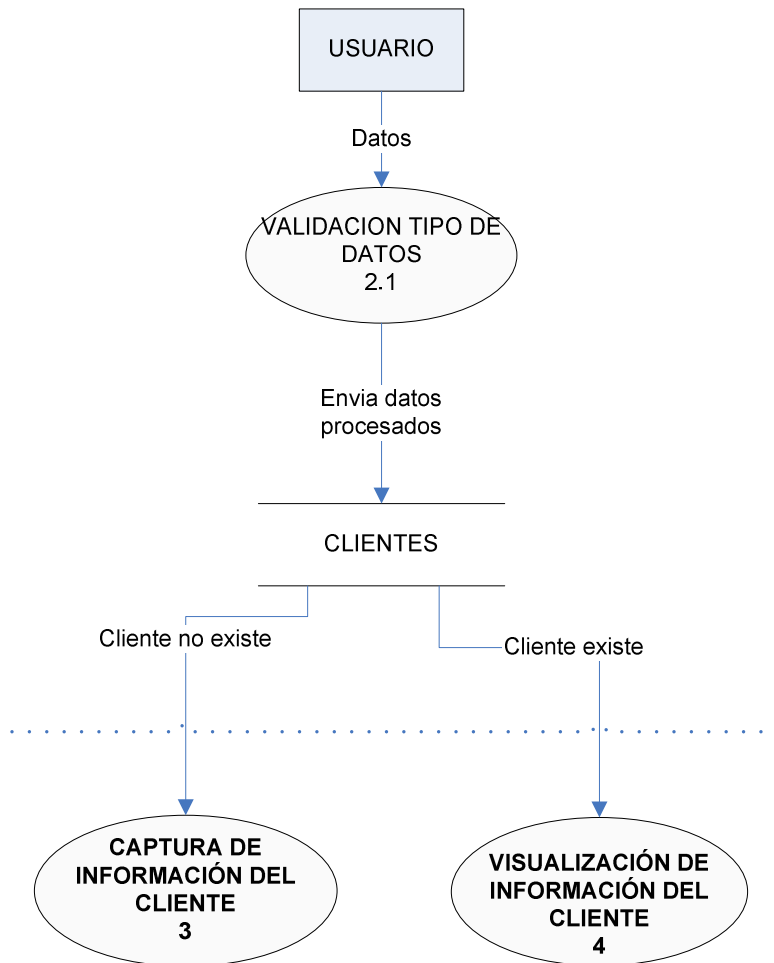
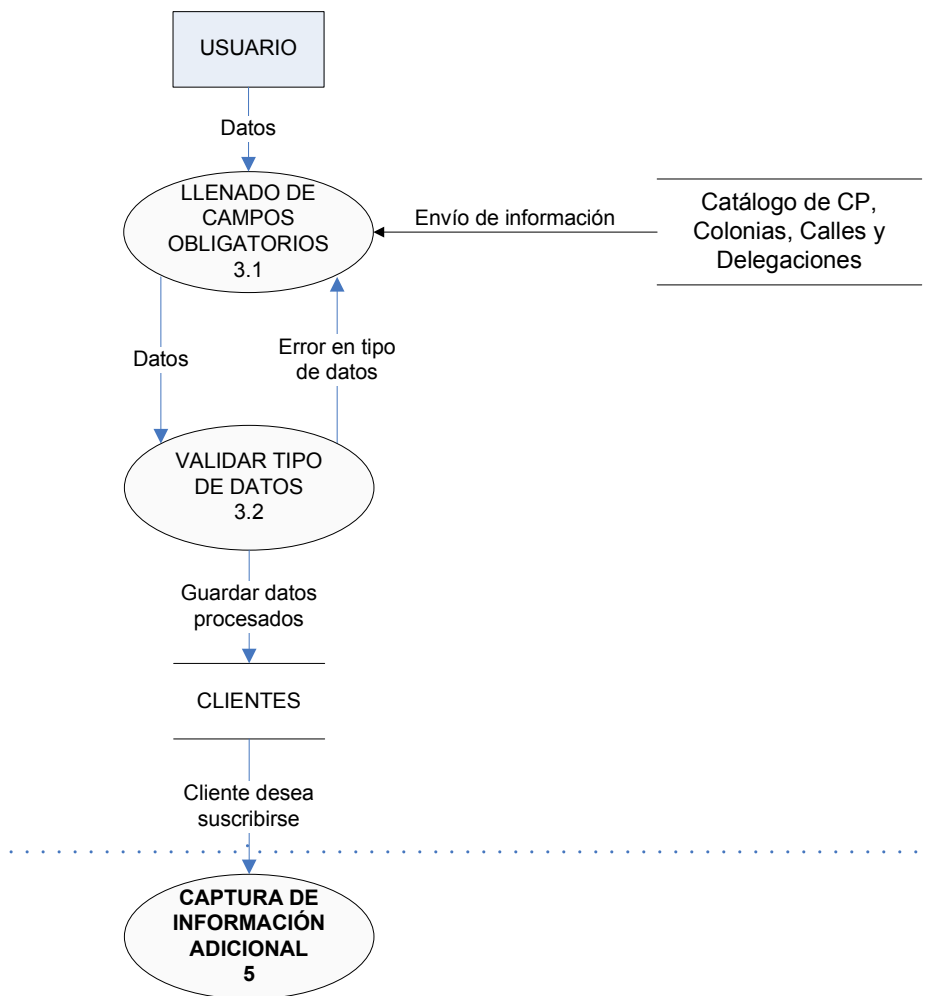


Figura 4.1.3.4 Búsqueda de clientes

Nuevamente se valida que los datos entregados por el usuario sean correctos y no afecten a la base de datos; si el cliente no se encuentra registrado se va al proceso de captura de información del cliente, en caso contrario, se pasa al proceso de visualización de información del cliente.



El DFD en segundo nivel para la captura de información del cliente se describe en la Figura 4.1.3.5

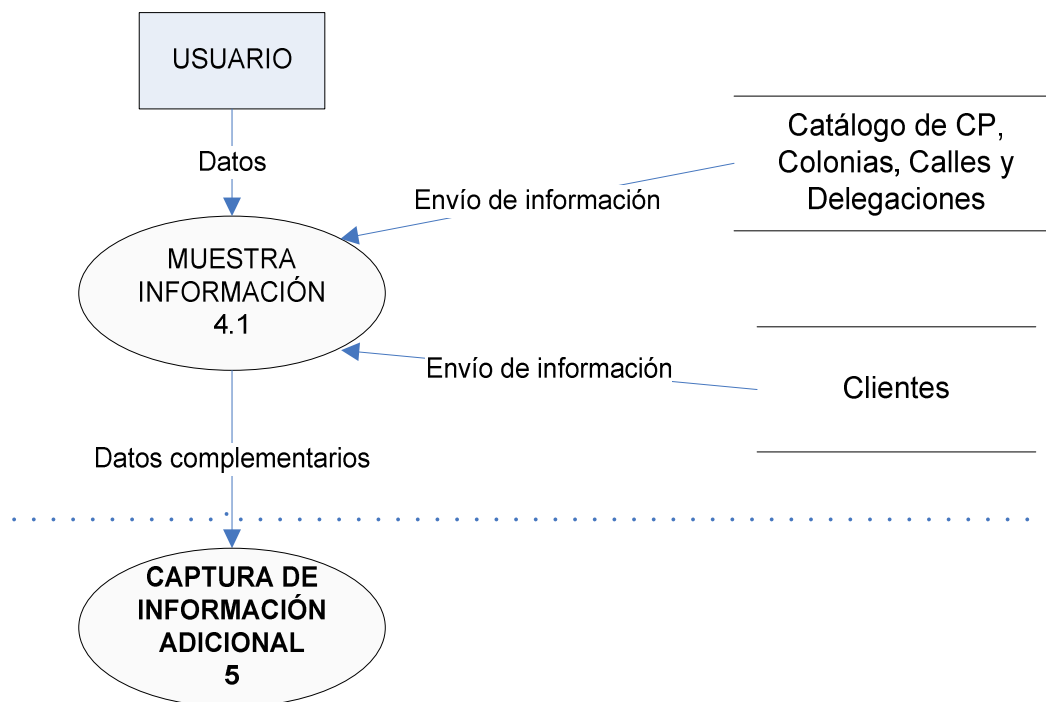


**Figura 4.1.3.5 Captura de información de clientes**

En la presente descomposición, el usuario envía los datos correspondientes del cliente interactuando con los catálogos de códigos postales, colonias, calles y delegaciones, posteriormente se valida el tipo de datos, y la información se guardará en la base de datos que almacenará los clientes, continuando con el proceso de captura de información adicional.



El DFD en su segundo nivel para la visualización de información del cliente se muestra en la Figura 4.1.3.6.



**Figura 4.1.3.6 Visualización de información del cliente**

En la presente descomposición, se muestra la información del cliente, obteniendo información de los catálogos de códigos postales, colonias, calles, delegaciones y de la base de datos que almacenará los clientes, continuando con el proceso de captura de información adicional.

El DFD en su segundo nivel para la captura de información adicional se muestra en la Figura 4.1.3.7



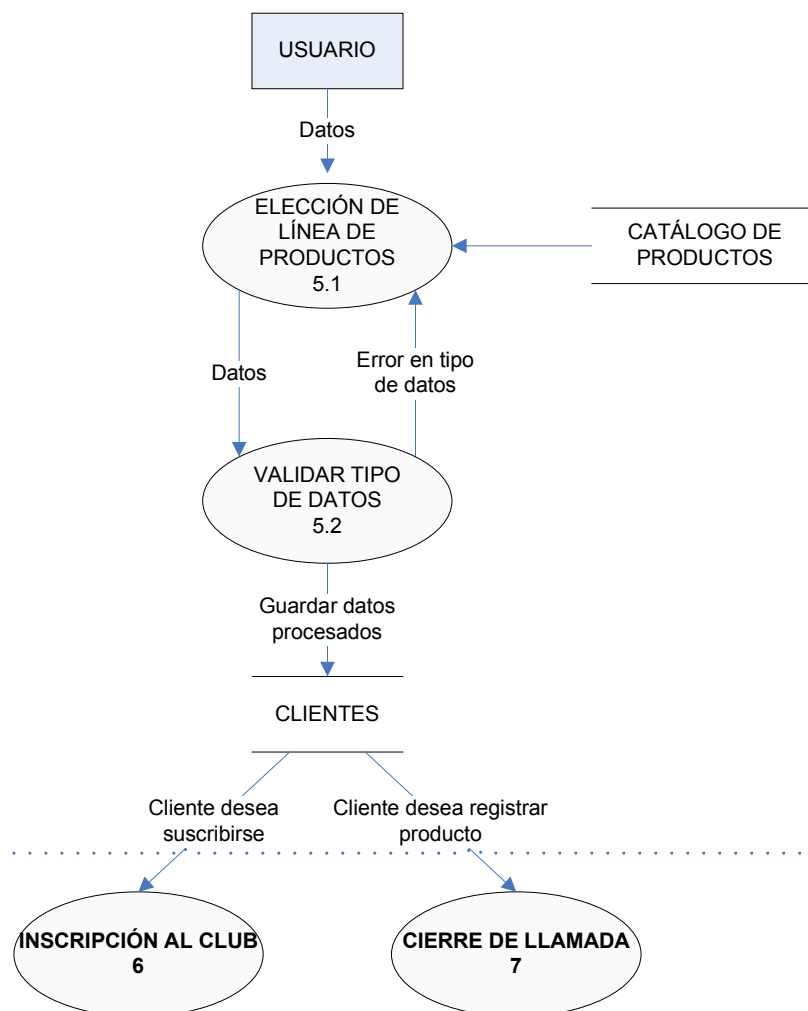


Figura 4.1.3.7 Captura de información adicional

En esta descomposición se obtienen del catálogo de productos, las líneas existentes de las cuales el usuario elige la proporcionada por el cliente, validándose el tipo de datos de los campos restantes, guardándose la información en la base de datos que almacenará los clientes. En caso de que el cliente desee inscribirse a algún club de la marca, se continúa con el proceso de Inscripción al club, en caso contrario, se pasa al proceso de Cierre de llamada.



El DFD en su segundo nivel para la inscripción al club se muestra en la Figura 4.1.3.8

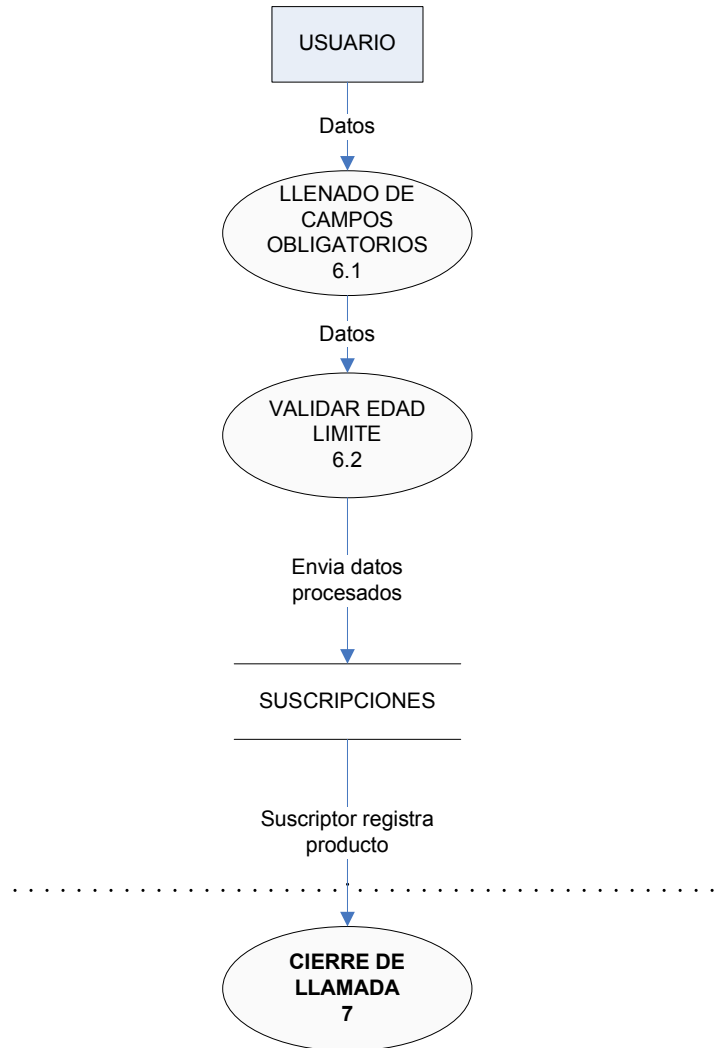


Figura 4.1.3.8 Inscripción al club

Como en los casos anteriores, el usuario envía los datos obligatorios del cliente, el proceso de Validar edad límite, verifica que el número de hijos del cliente no sobrepasen la edad de 4 años. Una vez hecho el registro en la base de datos, el cliente debe de pasar al proceso de Cierre de llamada, donde se clasifica la misma.



El DFD en segundo nivel para el cierre de llamada se presenta en la Figura 4.1.3.9.

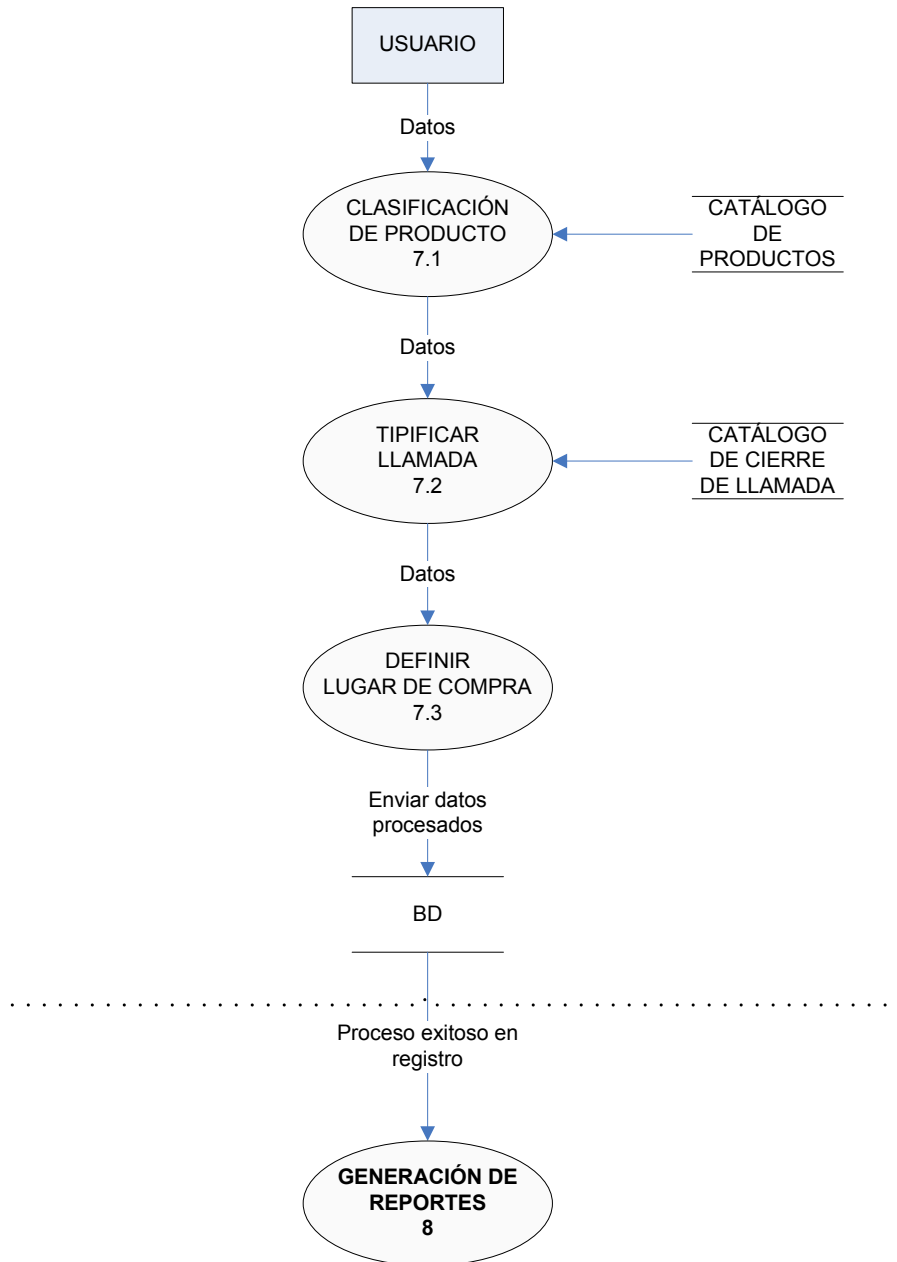
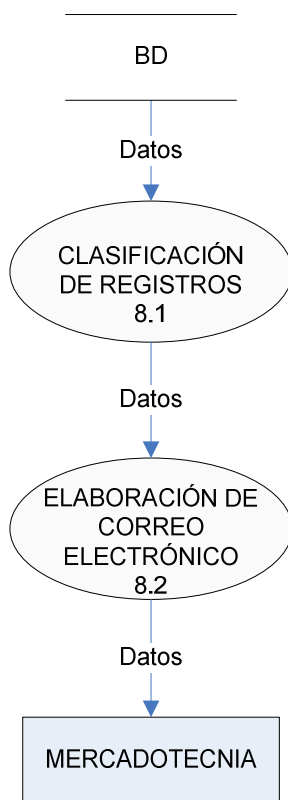


Figura 4.1.3.9 Cierre de llamada



En la descomposición de este proceso, el usuario realiza la clasificación de los productos por línea, se determina la tipificación de la llamada que se encuentra en el catálogo de cierre de llamadas dentro de la base de datos y se obtiene el lugar de compra de los productos que nos será de utilidad para la generación de reportes. Una vez hecho lo anterior, se guardan los registros en la base de datos y, finalmente, si todo el proceso tiene como respuesta proceso exitoso, pueden llevarse a cabo la generación de los reportes.

El DFD en segundo nivel para la generación de reportes se muestra en la Figura 4.1.3.10.



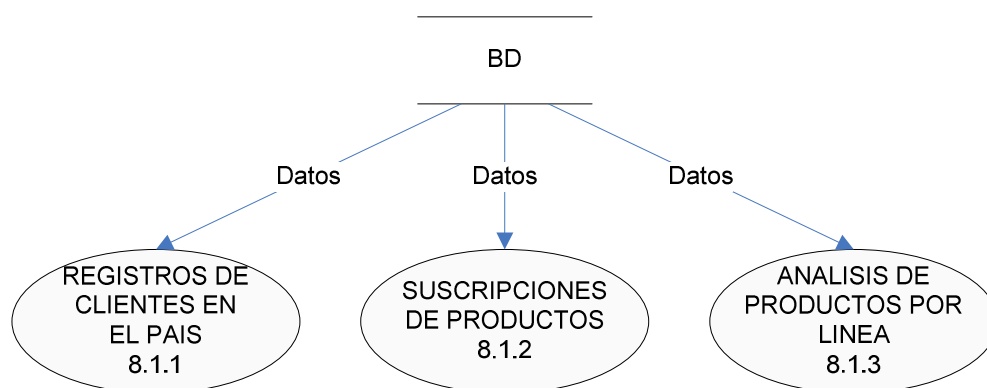
**Figura 4.1.3.10 Generación de reportes**



En este proceso, el flujo de toda la información proviene directamente de la base de datos. Existe un proceso complejo que es la clasificación de registros, una vez que los datos hayan pasado por éste, se realiza la elaboración del correo electrónico que se enviará al área de mercadotecnia.

Como mencionamos anteriormente, el proceso de clasificación de registros cuenta con procesos internos, por lo que se tiene la siguiente descomposición.

El DFD en su tercer nivel para la clasificación de registros se presenta en la Figura 4.1.3.11.



**Figura 4.1.3.11 Tercer nivel de clasificación de registros**

En este proceso se realiza la minería de datos de la información recabada por el sistema y, a petición del área de mercadotecnia, se generan reportes en cada uno de los procesos descritos. Ya con lo anterior se pasa al proceso de Elaboración de correo electrónico y se da por terminado el presente nivel.



### 4.1.4 Diccionario de datos

Es un conjunto de datos acerca de los datos (metadatos), que contiene las características lógicas de los mismos que se van a utilizar en el sistema, incluyendo nombre, descripción, alias, contenido y organización. En éste, se encuentra la lista de todos los elementos que forman parte del flujo de datos de todo el sistema.

En la Tabla 4.1.4.1 se tiene una breve descripción de las entidades y catálogos que se encuentran en el sistema, así como las relaciones que presentan con otras tablas.

Tabla	Descripción	Observaciones
CAMP_GRIZZI	Datos generales del cliente	Relación con tablas GRIZZI_CLIENTE_PRODUCTO (FK), GRIZZI_HIJO (FK) y GRIZZI_CLUB_RICITOS (FK)
GRIZZI_CLIENTE_PRODUCTO	Datos generales de los productos adquiridos por el cliente	Relación con tablas GRIZZI_HISTORICO_REPORTE(FK) y CAT_PRODUCTO_GRIZZI
GRIZZI_HIJO	Datos en numero de hijos y fechas de registro	Relación con tabla CAMP_GRIZZI
CAT_GRIZZI_DEPARTAMENTO	Catalogo de departamentos	Relación con tablas CAT_PRODUCTO_GRIZZI(FK) y CAT_GRIZZI GERENTE.
GRIZZI_CLUB_RICITOS	Datos generales de suscripción al club	Relación con tabla CAMP_GRIZZI
CAT_PRODUCTO_GRIZZI	Catalogo de productos	Relación con tablas CAT_GRIZZI_DEPARTAMENTO y GRIZZI_CLIENTE_PRODUCTO (FK)
GRIZZI_HISTORICO_REPORTE	Registro para elaboración de reportes	Relación con tabla GRIZZI_CLIENTE_PRODUCTO
CAT_GRIZZI GERENTE	Datos generales de gerente	Relación con tabla CAT_GRIZZI_DEPARTAMENTO
CIERRE_LLAMADA	Datos con descripción y tipos de cierre de llamada.	Tabla independiente
OTRA_LLAMADA	Registros no clasificados en CIERRE_LLAMADA	Tabla independiente

**Tabla 4.1.4.1 Descripción de tablas**



## Capítulo 4. Diseño e implementación

En las Tablas 4.1.4.2 a la 4.1.4.11, se presenta el diccionario de datos correspondiente al sistema

NOMBRE : CAMP_GRIZZI							
DESCRIPCION: Datos generales del cliente							
ACRONIMO	NOMBRE	LLAVE	TIPO DE DATOS	LON- GITUD	NULO	TABLA ASOCIADA	DESCRIPCION
iCampañaID	Clave campaña	√ (PK)	int	4	X	CAMP_GRIZZI	Clave consecutiva del cliente
cCampañaNombre	Nombre		varchar	30	√	CAMP_GRIZZI	Nombre(s) del cliente
cCampañaPaterno	Ap. Paterno		varchar	30	√	CAMP_GRIZZI	Apellido paterno del cliente
cCampañaMaterno	Ap. Materno		varchar	30	√	CAMP_GRIZZI	Apellido materno del cliente
cLada1	Lada 1		varchar	5	√	CAMP_GRIZZI	Clave lada principal
cTelefono1	Teléfono 1		varchar	10	√	CAMP_GRIZZI	Num. Telefónico principal
cLada2	Lada 2		varchar	5	√	CAMP_GRIZZI	Clave lada secundaria
cTelefono2	Teléfono 2		varchar	10	√	CAMP_GRIZZI	Num. Telefónico secundario
cCampañaCP	Código Postal		varchar	5	√	CAMP_GRIZZI	Código postal del cliente
cCampañaCalle	Calle		varchar	60	√	CAMP_GRIZZI	Calle del domicilio del cliente
cCampañaNumExt	Numero exterior		varchar	20	√	CAMP_GRIZZI	Num. Int. Del dom. del cliente
cCampañaNumInt	Numero interior		varchar	20	√	CAMP_GRIZZI	Num. Ext. Del dom. del cliente
cCampañaColonia	Colonia		varchar	50	√	CAMP_GRIZZI	Colonia del domicilio del cliente
cCampañaDelegacion	Delegación		varchar	50	√	CAMP_GRIZZI	Delegación o municipio del cliente
cCampañaCiudad	Ciudad		varchar	40	√	CAMP_GRIZZI	Ciudad del domicilio del cliente



## Capítulo 4. Diseño e implementación

cCampañaEstado	Estado		varchar	40	√	CAMP_GRIZZI	Estado del domicilio del cliente
cFechaNac	FechaNac		char	10	√	CAMP_GRIZZI	Fecha de nacimiento del cliente
cCorreoElectronico	Email		varchar	80	√	CAMP_GRIZZI	Email del cliente
cTieneHijos	TieneHijos		char	2	X	CAMP_GRIZZI	Dato afirmativo
iCuantosHijos	Cantidad de hijos		int	4	√	CAMP_GRIZZI	Numero de hijos que tiene el cliente
cOpinionProgramaRicitos	Opinión		varchar	255	√	CAMP_GRIZZI	Opinión de productos
cFechaAntigüedad	Fecha de antigüedad		varchar	10	√	CAMP_GRIZZI	Tiempo que lleva de conocer el producto
cSugerenciaRicitos	Sugerencias		varchar	255	√	CAMP_GRIZZI	Sugerencias para los productos
cInscribirse	Inscripción		char	2	√	CAMP_GRIZZI	Dato afirmativo
bInscrito	Estatus		bit	1	√	CAMP_GRIZZI	Estatus de inscripción
fiAgenteID	AgenteID		int	4	√	CAMP_GRIZZI	Clave del operador telefónico
dFechaRegistro	Fecha		smalldatetime	4	√	CAMP_GRIZZI	Fecha de registro del cliente

**Tabla 4.1.4.2 Entidad CAMP\_GRIZZI**

NOMBRE : CAT_GRIZZI_DEPARTAMENTO							
DESCRIPCION: Catalogo de departamentos							
ACRONIMO	NOMBRE	LLAVE	TIPO DE DATOS	LON-GITUD	NULO	TABLA ASOCIADA	DESCRIPCION
iDeptoID	Deptold	√ (PK)	int	4	X	CAT_GRIZZI_DEPARTAMENTO	Clave de departamento
cDeptoDesc	Descripción		varchar	50	X	CAT_GRIZZI_DEPARTAMENTO	Descripción de departamento
bDeptoEstatus	Estatus		bit	1	X	CAT_GRIZZI_DEPARTAMENTO	Estatus actual del departamento

**Tabla 4.14.3 Entidad CAT\_GRIZZI\_DEPARTAMENTO**





NOMBRE : GRIZZI_CLIENTE_PRODUCTO							
DESCRIPCION: Datos generales de los productos adquiridos por el cliente							
ACRONIMO	NOMBRE	LLAVE	TIPO DE DATOS	LON-GITUD	NULO	TABLA ASOCIADA	DESCRIPCION
iProductoCte	ProductoCliente	√ (PK)	int	4	X	GRIZZI_CLIENTE_PRODUCTO	Clave principal cliente _ producto
iCampañaID	Cve campaña	√ (FK)	int	4	X	CAMP_GRIZZI	Clave consecutiva del cliente
iGrizziID	Cve producto	√ (FK)	int	4	X	CAT_PRODUCTO_GRIZZI	Clave de productos
iGrizziLinea	Cve Línea		int	4	√	GRIZZI_CLIENTE_PRODUCTO	Clave de línea de productos
iGrizziFamilia	Cve Familia		int	4	√	GRIZZI_CLIENTE_PRODUCTO	Clave de familia de productos
cLugarCompra	LugarCompra		varchar	40	√	GRIZZI_CLIENTE_PRODUCTO	Lugar de compra de los productos
iCierreID	CierreID		int	4	√	GRIZZI_CLIENTE_PRODUCTO	Tipo de cierre de llamada
cComentarios	Comentarios		varchar	1000	√	GRIZZI_CLIENTE_PRODUCTO	Comentarios de los productos
fiAgenteID	AgenteID		int	4	√	GRIZZI_CLIENTE_PRODUCTO	Clave del operador telefónico
dFechaRegistro	Fecha		small datetime	4	√	GRIZZI_CLIENTE_PRODUCTO	Fecha de registro de los productos

Tabla 4.1.4.4 Entidad GRIZZI\_CLIENTE\_PRODUCTO



## Capítulo 4. Diseño e implementación

NOMBRE : GRIZZI_CLUB_RICITOS							
DESCRIPCION: Datos generales de suscripción al club							
ACRONIMO	NOMBRE	LLAVE	TIPO DE DATOS	LONGITUD	NULO	TABLA ASOCIADA	DESCRIPCION
iClubID	ClubId	√ (PK)	int	4	X	GRIZZI_CLUB_RICITOS	Clave de la membresía
iCampañaID	Clave campaña	√ (FK)	int	4	X	CAMP_GRIZZI	Clave consecutiva del cliente
bNacido	Nacido		bit	1	X	GRIZZI_CLUB_RICITOS	Estatus de nacimiento
cFechaNacimiento	FechaNac		varchar	10	√	GRIZZI_CLUB_RICITOS	Fecha de nacimiento del bebé
cBebeNombre	BebeNombre		varchar	30	√	GRIZZI_CLUB_RICITOS	Nombre(s) del bebé
cBebePaterno	Ap. Paterno		varchar	30	√	GRIZZI_CLUB_RICITOS	Apellido paterno del bebé
cBebeMaterno	Ap. Materno		varchar	30	√	GRIZZI_CLUB_RICITOS	Apellido materno del bebé
CSexo	Sexo		char	1	√	GRIZZI_CLUB_RICITOS	Sexo del bebé
cPapaNombre	PapaNombre		varchar	30	√	GRIZZI_CLUB_RICITOS	Nombre(s) del padre
cPapaPaterno	PapaPaterno		varchar	30	√	GRIZZI_CLUB_RICITOS	Apellido paterno del padre
cPapaMaterno	PapaMaterno		varchar	30	√	GRIZZI_CLUB_RICITOS	Apellido materno del padre
fiAgenteID	AgentId		int	4	√	GRIZZI_CLUB_RICITOS	Clave del operador telefónico
dFechaRegistro	Fecha		smalldatetime	4	√	GRIZZI_CLUB_RICITOS	Fecha de registro de los datos del bebé

**Tabla 4.1.4.5 Entidad GRIZZI\_CLUB\_RICITOS**

NOMBRE : GRIZZI_HIJO							
DESCRIPCION: Datos en numero de hijos y fechas de registro							
ACRONIMO	NOMBRE	LLAVE	TIPO DE DATOS	LONGITUD	NULO	TABLA ASOCIADA	DESCRIPCION
iHijoID	HijoID	√ (PK)	int	4	X	GRIZZI_HIJO	Clave consecutiva del hijo



## Capítulo 4. Diseño e implementación

iCampañaID	Clave campaña	√ (FK)	int	4	X	CAMP_GRIZZI	Clave consecutiva del cliente
iEdadHijo	Edad		int	4	X	GRIZZI_HIJO	Edad del hijo
iHijoNumero	HijoNumero		int	4	X	GRIZZI_HIJO	Cantidad de hijos
dFechaRegistro	Fecha		smalldatetime	4	X	GRIZZI_HIJO	Fecha de registro de los datos del hijo

**Tabla 4.1.4.6 Entidad GRIZZI\_HIJO**

NOMBRE : GRIZZI_HISTORICO_REPORTE							
DESCRIPCION: Registro para elaboración de reportes							
ACRONIMO	NOMBRE	LLAVE	TIPO DE DATOS	LON-GITUD	NULO	TABLA ASOCIADA	DESCRIPCION
iHistoricoID	HistoricoID	√ (PK)	int	4	X	GRIZZI_HISTORICO_REPORTE	Clave de bitácora
iProductoCte	ProductoCliente	√ (FK)	int	4	X	GRIZZI_CLIENTE_PRODUCTO	Clave principal cliente _ producto
fiAgentelD	AgentelD		int	4	√	GRIZZI_HISTORICO_REPORTE	Clave del operador telefónico
dFechaRegistro	Fecha		smalldatetime	4	X	GRIZZI_HISTORICO_REPORTE	Fecha de registro de bitácora

**Tabla 4.1.4.7 Entidad GRIZZI\_HISTORICO\_REPORTE**

NOMBRE : CAT_PRODUCTO_GRIZZI							
DESCRIPCION: Catalogo de productos							
ACRONIMO	NOMBRE	LLAVE	TIPO DE DATOS	LONGITUD	NULO	TABLA ASOCIADA	DESCRIPCION
iGrizziID	Clave producto	√ (PK)	int	4	X	CAT_PRODUCTO_GRIZZI	Clave de productos
iDeptolD	DeptolD	√ (FK)	int	4	X	CAT_GRIZZI_DEPARTAMENTO	Clave de departamento
iGrizziEsMiPadre	GrizziPadre		int	4	√	CAT_PRODUCTO_GRIZZI	Clave para recursividad en tabla
cGrizziProducto	GrizziProducto		varchar	80	X	CAT_PRODUCTO_GRIZZI	Nombre del producto
bGrizziEstatus	Estatus		bit	1	X	CAT_PRODUCTO_GRIZZI	Estatus del producto

**Tabla 4.1.4.8 Entidad CAT\_PRODUCTO\_GRIZZI**



NOMBRE : CAT_GRIZZI_GERENTE							
DESCRIPCION: Datos generales de gerente							
ACRONIMO	NOMBRE	LLAVE	TIPO DE DATOS	LON-GITUD	NULO	TABLA ASOCIADA	DESCRIPCION
iGerenteID	GerenteID	√ (PK)	int	4	X	CAT_GRIZZI_GERENTE	Clave para definir gerencias
iDeptoID	DeptoID	√ (FK)	int	4	X	CAT_GRIZZI_DEPARTAMENTO	Clave de departamento
cGerenteNombre	GerenteNombre		varchar	90	X	CAT_GRIZZI_GERENTE	Nombre(s) completos del gerente
cGerenteCorreo	GerenteCorreo		varchar	50	X	CAT_GRIZZI_GERENTE	Email del gerente
cGerenteTelefono	Teléfono		varchar	20	√	CAT_GRIZZI_GERENTE	Teléfono de contacto
cGerenteExt	Extensión		varchar	20	√	CAT_GRIZZI_GERENTE	Extensión telefónica
cDepartamento	Departamento		varchar	50	√	CAT_GRIZZI_GERENTE	Departamento de gerencia
bGerenteEstatus	Estatus		bit	1	X	CAT_GRIZZI_GERENTE	Estatus del gerente
dFechaRegistro	Fecha de registro		small datetime	4	√	CAT_GRIZZI_GERENTE	Fecha de registro del gerente

Tabla 4.1.4.9 Entidad CAT\_GRIZZI\_GERENTE

NOMBRE : CIERRE_LLAMADA							
DESCRIPCION: CIERRE_LLAMADA							
ACRONIMO	NOMBRE	LLAVE	TIPO DE DATOS	LONGITUD	NULO	TABLA ASOCIADA	DESCRIPCION
iCierreID	CierreID	√ (PK)	int	4	X	CIERRE_LLAMADA	Clave del tipo de cierre
cCierreDescripcion	CierreDescripcion		varchar	30	√		Descripción de los cierres
bCierreStatus	Estatus		bit	1	X		Estatus del tipo de cierre

Tabla 4.1.4.10 Entidad CIERRE\_LLAMADA



NOMBRE : OTRA_LLAMADA							
DESCRIPCION: Registros no clasificados en CIERRE_LLAMADA							
ACRONIMO	NOMBRE	LLAVE	TIPO DE DATOS	LON-GITUD	NULO	TABLA ASOCIADA	DESCRIPCION
iOtraLlamadaID	OtraLlamadaID	√ (PK)	int	4	X	OTRA_LLAMADA	Clave consecutiva para otro tipo de cierre
iCierreID	CierreID		int	4	√	OTRA_LLAMADA	Clave del tipo de cierre
fiAgentelD	AgentelD		int	4	√	OTRA_LLAMADA	Clave del operador telefónico
cOtraLlamadaObs	cOtraLlamadaObs		varchar	255	√	OTRA_LLAMADA	Descripción para otro tipo de cierre
dFechaRegistro	Fecha		small datetime	4	√	OTRA_LLAMADA	Fecha de registro para otro tipo de cierre

Tabla 4.1.4.11 Entidad OTRA\_LLAMADA

### 4.1.5 Diagrama entidad-relación (DER)

Es una herramienta de modelado para bases de datos, mediante la cual se pretenden visualizar los objetos que pertenecen a la base de datos como entidades, las cuales, tienen atributos y se vinculan mediante relaciones.

Aplicando la teoría antes descrita para la creación de los diagramas entidad-relación, es posible obtener el siguiente diagrama entidad-relación descrito en la Figura 4.1.5.1 en su diseño lógico.

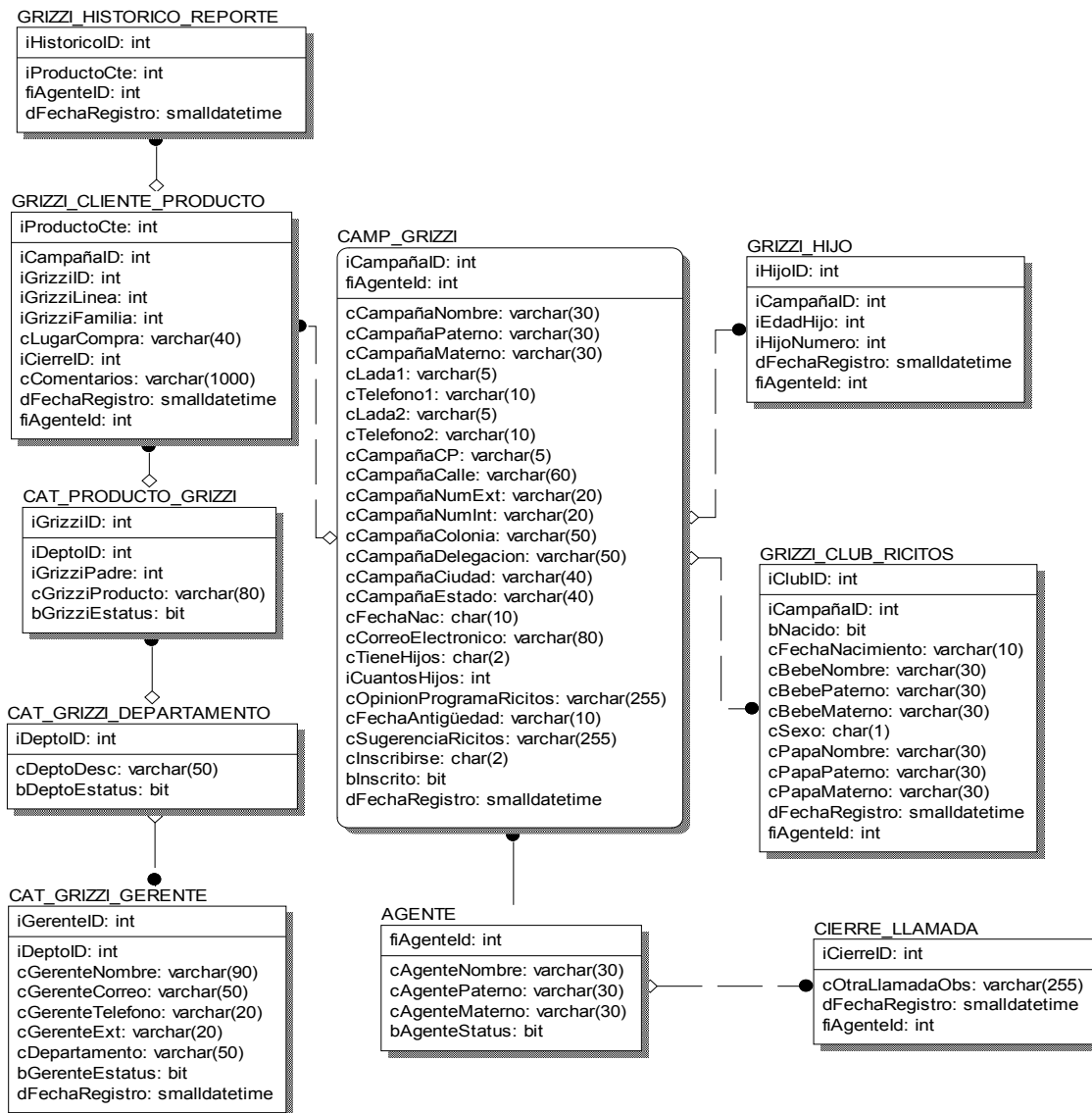


Figura 4.1.5.1 Diagrama entidad-relación, diseño lógico

#### 4.1.6 Normalización de la base de datos

Los motivos por los cuales se debe de llevar a cabo la normalización en las tablas de la base de datos son los siguientes:



- Minimiza la redundancia de los datos; un dato sin normalizar es redundante.
- La redundancia de datos causa problemas de integridad. Las transacciones de actualización y borrado pueden no ser consistentes en todas las copias de los datos causando inconsistencia en la base de datos.
- La normalización ayuda a identificar entidades, relaciones y tablas faltantes.

A continuación se realiza el análisis de normalización de nuestra base de datos, tomando como muestra algunos de los datos obtenidos de la información dentro del call center.

Al examinar los registros de la Tabla 4.1.6.1, podemos darnos cuenta que contienen un grupo repetido para GRIZZI\_ITEM, LUGAR\_COMPRA, CANT y COMENTARIOS.

ID_CAMPAÑA	FECHA	ID_CLIENTE	NOM_CLIENTE	ESTADO	GRIZZI_ITEM	LUGAR_COMPRA	CANT	COMENTARIOS
2301	12/04/2004	101	ANGELES PATLAN RODRIGUEZ	ESTADO DE MEXICO	256	MEGA COMERCIAL PILARES	3	35
2301	12/04/2004	101	ANGELES PATLAN RODRIGUEZ	ESTADO DE MEXICO	311	MEGA COMERCIAL PILARES	6	65
2301	12/04/2004	101	ANGELES PATLAN RODRIGUEZ	ESTADO DE MEXICO	311	MEGA COMERCIAL PILARES	8	4.75
2302	29/04/2004	107	ALMA ROSA LOPEZ ANDA	ESTADO DE MEXICO	560	FARMACIAS	4	5
2303	30/04/2004	110	MAYRA ROMERO AGUILERA	DISTRITO FEDERAL	4011	FARMACIAS	2	65
2303	04/05/2004	110	VICENTE CABRERA PEREZ	JALISCO	3141	HEB	2	10

**Tabla 4.1.6.1 Datos de muestra**

La primera forma normal (1FN) prohíbe los grupos repetidos, por lo tanto tenemos que hacer que cumpla con este criterio. Los pasos a seguir son:

- Eliminar los grupos repetidos.



- Crear una nueva tabla con la llave primaria de la tabla base y el grupo repetido.

Los registros quedan ahora conformados en dos tablas que llamaremos CAMP\_GRIZZI (Ver Tabla 4.1.6.2) y GRIZZI\_CLIENTE\_PRODUCTO (Ver Tabla 4.1.6.3).

ID_CAMPAÑA	FECHA	ID_CLIENTE	NOM_CLIENTE	ESTADO
2301	12/04/2004	101	ANGELES PATLAN RODRIGUEZ	ESTADO DE MEXICO
2301	12/04/2004	101	ANGELES PATLAN RODRIGUEZ	ESTADO DE MEXICO
2301	12/04/2004	101	ANGELES PATLAN RODRIGUEZ	ESTADO DE MEXICO
2302	29/04/2004	107	ALMA ROSA LOPEZ ANDA	ESTADO DE MEXICO
2303	30/04/2004	110	MAYRA ROMERO AGUILERA	DISTRITO FEDERAL
2303	04/05/2004	110	VICENTE CABRERA PEREZ	JALISCO

**Tabla 4.1.6.2 CAMP\_GRIZZI**

ID_CAMPAÑA	GRIZZI_ITEM	LUGAR_COMPRA	CANT	COMENTARIOS
2301	256	MEGA COMERCIAL PILARES	3	NINGUNO
2301	311	MEGA COMERCIAL PILARES	6	NINGUNO
2301	311	MEGA COMERCIAL PILARES	8	NINGUNO
2302	560	FARMACIAS	4	NINGUNO
2303	4011	FARMACIAS	2	NINGUNO
2303	3141	HEB	2	NINGUNO

**Tabla 4.1.6.3 GRIZZI\_CLIENTE\_PRODUCTO**

Ahora procederemos a aplicar la segunda forma normal (2FN), es decir, tenemos que eliminar cualquier columna no llave que no dependa de la llave primaria de la tabla. Los pasos a seguir son:





- Determinar cuáles columnas que no son llave no dependen de la llave primaria de la tabla.
- Eliminar esas columnas de la tabla base.
- Crear una segunda tabla con esas columnas y la(s) columna(s) de la llave primaria (PK) de la cual dependen.

La tabla CAMP\_GRIZZI está en 2FN. Cualquier valor único de ID\_CAMPANA determina un sólo valor para cada columna. Por lo tanto, todas las columnas son dependientes de la llave primaria ID\_CAMPANA.

Por su parte, la tabla GRIZZI\_CLIENTE\_PRODUCTO no se encuentra en 2FN ya que las columnas LUGAR\_COMPRA y COMENTARIOS son dependientes de GRIZZI\_ITEM, pero no son dependientes de ID\_ORDEN.

Lo que haremos a continuación es eliminar estas columnas de la tabla GRIZZI\_CLIENTE\_PRODUCTO y crear una tabla PRODUCTOS con dichas columnas y la llave primaria de la que dependen. El resultado se muestra en las tablas 4.1.6.4 y 4.1.6.5.

ID_CLIENTE	NOM_CLIENTE	ESTADO	FECHA
101	ANGELES PATLAN RODRIGUEZ	ESTADO DE MEXICO	12/04/2004
101	ANGELES PATLAN RODRIGUEZ	ESTADO DE MEXICO	12/04/2004
101	ANGELES PATLAN RODRIGUEZ	ESTADO DE MEXICO	12/04/2004
107	ALMA ROSA LOPEZ ANDA	ESTADO DE MEXICO	29/04/2004
110	MAYRA ROMERO AGUILERA	DISTRITO FEDERAL	30/04/2004
110	VICENTE CABRERA PEREZ	JALISCO	04/05/2004

**Tabla 4.1.6.4 CAMP\_GRIZZI**



GRIZZI_ITEM	LUGAR_COMPRA	CANT	COMENTARIOS
256	MEGA COMERCIAL PILARES	3	NINGUNO
311	MEGA COMERCIAL PILARES	6	NINGUNO
311	MEGA COMERCIAL PILARES	8	NINGUNO
560	FARMACIAS	4	NINGUNO
4011	FARMACIAS	2	NINGUNO
3141	HEB	2	NINGUNO

**Tabla 4.1.6.5 GRIZZI\_CLIENTE\_PRODUCTO**

La tercera forma normal (3FN) nos dice que tenemos que eliminar cualquier columna no llave que sea dependiente de otra columna no llave. Los pasos a seguir son:

- Determinar las columnas que son dependientes de otra columna no llave.
- Eliminar esas columnas de la tabla base.
- Crear una segunda tabla con esas columnas y con la columna no llave de la cual son dependientes.

Al observar las Tablas 4.1.6.4 y 4.1.6.5, nos damos cuenta que tanto la tabla CAMP\_GRIZZI, como la tabla GRIZZI\_CLIENTE\_PRODUCTO se encuentran en 3FN.

## 4.2. DESARROLLO DE LA INTERFAZ DEL USUARIO (FRONT-END)

La aplicación tiene como inicio la pantalla de acceso al sistema en donde se muestran dos cajas de texto (textbox), una para colocar el nombre del usuario y la otra para la contraseña, también se muestra un botón (button) llamado **Entrar**, el



cual se encargará de enviar los datos que fueron colocados en los dos campos anteriores. (Ver Figura 4.2.1).

En la parte superior derecha de la pantalla existe otro botón, el cual nos permitirá cerrar la ventana del explorador, no sin antes enviar un mensaje confirmando el cierre de la misma.

La imagen muestra una ventana de software con un título azul oscuro que contiene el botón "CERRAR" en la esquina superior derecha. El contenido principal de la ventana es un formulario de inicio de sesión con el texto "Por favor ingrese la siguiente información para acceder a la aplicación." centrado. Debajo de este texto hay dos campos de entrada: "Usuario:" y "Password:", cada uno con un cuadro de texto rectangular. Debajo de los campos de texto hay un botón rectangular con el texto "ENTRAR".

**Figura 4.2.1 Pantalla de acceso al sistema**

Cuando se ha ingresado un usuario válido para el sistema, se desplegará la pantalla de búsqueda del cliente, en donde aparecerán los campos No. Contacto, Nombre, Apellido Paterno y Teléfono, cada uno con una caja de texto del lado derecho y que estarán contenidos dentro de un marco (field set), que agrupa los campos que pertenecen a la sección de búsqueda del cliente, esta pantalla también contendrá un botón llamado **Buscar** (Ver Figura 4.2.2).



Búsqueda del Cliente

No. Contacto :

Nombre :  A. Paterno :

Teléfono :

BUSCAR

CERRAR

**Figura 4.2.2 Pantalla de búsqueda del cliente**

Tecleando los datos correspondientes en los campos anteriores y dando click en el botón **Buscar**, se utiliza el stored procedure llamado `pr_GRBusca_Cliente`, que busca y despliega en una tabla (table) las coincidencias encontradas. En caso de no existir ningún registro, se enviará un mensaje de error indicando que no existe ningún cliente con la información proporcionada. (Ver Figura 4.2.3).

Búsqueda del Cliente

No. Contacto :

Nombre :  A. Paterno :

Teléfono :

NUEVO REGISTRO

BUSCAR

Resultados de la búsqueda

No. Cte.	Nombre	Teléfono	Estado	Inscrito
1	Alejandro Fiesco	(223) 25123685	Puebla	No
2	Araceli Velazquez	(223) 25123685	Puebla	No
3	Alejandro Fiesco	(223) 25123685	Puebla	No
4	Araceli Velazquez	(223) 25123685	Puebla	No
5	Alejandro Fiesco	(223) 25123685	Puebla	No

**Figura 4.2.3 Resultados de coincidencias en la búsqueda**



Para realizar el alta de un nuevo cliente, se utilizará un botón llamado **Nuevo registro**. En todas las pantallas del sistema, se tendrá un botón en la parte superior de la pantalla para cerrar la ventana del explorador. Estas funciones interactúan con la tabla (CAMP\_GRIZZI).

En la parte inferior de la pantalla, se muestra un marco llamado Cierres de llamada, dentro del cual se encuentran una etiqueta (label), una lista desplegable (combobox) con varias opciones para el cierre y un botón. Este proceso interactúa con la tabla (CIERRE\_LLAMADA).

En caso de agregar un cliente nuevo, se despliega la pantalla de datos con todos los campos disponibles para capturar el nuevo registro (Ver Figura 4.2.4).

La imagen muestra una interfaz de usuario para el registro de un nuevo cliente. El formulario está dividido en varias secciones:

- Teléfono:** Campos para Teléfono1 (322 - 2994875), Teléfono2 (322 - 2994875) y E-mail.
- Cliente:** Campos para Nombre (ABEL), Ap. Paterno (SAGREO), Ap. Materno (REYES) y Fecha de Nac. (12/01/1978).
- Domicilio:** Campos para C.P. (48315), Estado (JALISCO), Colonia (COL. ARBOLEDAS), Del. Mun. (PUERTO VALLARTA), Ciudad (PUERTO VALLARTA), Calle (CONCHACHINAS), No. Ext. (533) y No. Int.
- Datos Adicionales:** Incluye un campo para Línea, botones de radio para "¿Desea Inscribirse?" (Si/No) y "¿Tiene hijos?" (Si/No), un campo para Fecha Antigüedad (01/05/2001), un campo para No. de Hijos (01), un menú desplegable para Opinión de Ricios (Muy Buena, Buena, Regular, Mala, Muy Mala, NS/NC, Ninguna) y un campo para Sugerencia.
- Cierres de Llamada:** Un menú desplegable para Tipo de cierre y un botón para REGISTRAR LLAMADA.

En la parte superior derecha del formulario hay un botón CERRAR. En la parte inferior del formulario hay un botón GUARDAR.

Figura 4.2.4 Pantalla de captura de datos



Esta pantalla está dividida en cinco secciones delimitadas mediante marcos, cada una con su respectivo nombre:

**Teléfono.** Contiene dos campos: el número de casa junto con su clave lada y el adicional el cual puede ser el número de oficina, celular o segunda línea de casa y si el cliente tiene cuenta de correo también existe el campo para este dato, todos estos con cajas de texto del lado derecho y colocados en la parte superior de la pantalla.

**Cliente.** En esta sección existen los campos nombre, apellido paterno y apellido materno, los tres con sus respectivas cajas de texto colocadas a la derecha de las etiquetas, el campo de fecha de nacimiento se podrá seleccionar a través de tres listas desplegables uno para día, otro para mes y año.

**Domicilio.** En esta sección se capturará el c.p. a través de una caja de texto y dependiendo de la información capturada en dicho campo, se desplegarán los datos de colonia, delegación/municipio, ciudad y estado en listas desplegables, los datos de calle, no. exterior e interior se capturarán en cajas de texto. En este proceso se ejecuta el stored procedure (pr\_GRInserta\_Cliente) y se trabaja con la tabla (CAMP\_GRIZZI).

**Datos adicionales.** El primer dato que se despliega es la línea a la cual pertenece cada uno de los productos que se compraron, esto se hace por medio de una lista desplegable, la cual despliega siete opciones, este campo trabaja con el stored procedure (pr\_GRConsulta\_Producto) y con la tabla (CAT\_GRIZZI\_DEPARTAMENTO).

El siguiente campo corresponde al de preguntar si se quiere inscribir al club del producto en campaña, esto se realiza por medio de dos botones de opción (radio button) **-Si y No-**.

---



Después se tiene otro campo para preguntar si el cliente tiene hijos y también se maneja a través de botones de opción.

Otra decisión que existe en esta sección es la opinión que tiene el cliente del producto específico de campaña, esto se lleva a cabo por medio de una lista desplegable, la cual contiene varias opciones para describirla.

Se tendrá un campo para introducir el tiempo que el cliente tiene de conocer los productos que mencionó en el apartado de línea utilizando una lista desplegable para cada campo de la fecha (día, mes, año), otro campo que se capturará a través de una caja de texto será el de número de hijos.

Para este caso se trabajará con la tabla (GRIZZI\_HIJO) y por último se tendrá un campo para capturar alguna sugerencia sobre los productos o servicios que ofrece la marca, para la cual se colocó un área de texto (textarea) con capacidad de hasta 255 caracteres.

Al finalizar la captura de todos los campos, se procede a guardar los datos a través de un botón llamado **Guardar**, este proceso trabaja con las tablas (GRIZZI\_CLIENTE\_PRODUCTO) y (CAMP\_GRIZZI).

Si en la pantalla de datos se eligió la opción de inscribirse al club de la campaña, en el momento en que se guarda el registro, se cambia a la pantalla de inscripción. (Ver Figura 4.2.5.)



FAQ | BUSCAR C.P. | SALIR | CERRAR

**Datos Conyuge**

Nombre:  Ap. Paterno:   
Ap. Materno:

**Datos Bebé**

Nacido  Por Nacer

Nombre:  Ap. Paterno:   
Ap. Materno:   
Fecha de Nac.:    Sexo:  Femenino  Masculino

**Otros Hijos**

Edad	¿Cuántos Hijos?
<input type="checkbox"/> 1	<input type="text"/>
<input type="checkbox"/> 2	<input type="text"/>
<input type="checkbox"/> 3	<input type="text"/>
<input type="checkbox"/> 4	<input type="text"/>
<input type="checkbox"/> 5	<input type="text"/>
<input type="checkbox"/> 6	<input type="text"/>
<input type="checkbox"/> De 7 a 10	<input type="text"/>
<input type="checkbox"/> Mas de 10	<input type="text"/>

**Cierres de Llamada**

Tipo de cierre:

**Figura 4.2.5 Pantalla de inscripción al club**

Esta pantalla está dividida en cuatro secciones utilizando marcos para agrupar cada una de ellas. En la primera, se tendrán que introducir los datos del cónyuge (nombre, apellido paterno y apellido materno) por medio de una caja de texto para cada uno de los campos, éstos se ubican en la parte superior de la pantalla.

En la siguiente sección, se introducirán los datos del bebé que se va a inscribir, por medio de seis campos, en el primero por medio de dos botones de opción se podrá inscribir a un bebé que ya nació o a un bebé que está por nacer, en los siguientes tres campos por medio de cajas de texto se podrán introducir el nombre, apellido paterno y materno del bebé.

A continuación se solicita la fecha de nacimiento del bebé para el cual se asignaron tres listas desplegables que presentarán los días del mes, los meses del





año y los años que estén dentro del margen de la edad estipulada para pertenecer al club y, por último, en el siguiente campo se debe de colocar el sexo del bebé, esto se realiza por medio de dos botones de opción, uno para femenino y otro para masculino.

En la sección “Otros hijos” se podrán capturar la edad y el número de hijos en caso de que el cliente tenga otros hijos a parte del que se esta inscribiendo. Esto se realiza por medio de cajas de texto y cajas de lista (check box). Esta sección trabajará con la tabla (GRIZZI\_HIJO). La pantalla estará ejecutando el stored procedure pr\_GRInserta\_al\_Club y trabajará con la tabla GRIZZI\_CLUB\_RICITOS cuando se presione el botón de **Guardar** al final de la página.

Independientemente de que se haya inscrito o no al club se llegará a la pantalla de clasificación de llamada (Ver Figura 4.2.6).

La imagen muestra una interfaz de usuario para la clasificación de llamadas. En la parte superior hay una barra de menú con los botones: FAQ, BUSCAR C.P., SALIR y CERRAR. El formulario principal, titulado "Clasificación de la Llamada", contiene los siguientes campos:

- Línea: campo de lista desplegable.
- Producto: campo de lista desplegable.
- Subproducto: campo de lista desplegable.
- Tipificación: campo de lista desplegable.
- Lugar de Compra: campo de texto.
- Comentarios: campo de texto con barra de desplazamiento vertical.

Debajo de estos campos hay un botón "GUARDAR".

En la parte inferior del formulario, hay un campo "Cierres de Llamada" con un subcampo "Tipo de cierre:" y un botón "REGISTRAR LLAMADA".

**Figura 4.2.6** Pantalla de clasificación de llamada



En la sección de **Clasificación de la llamada** se utilizan varias listas desplegables para presentar la información correspondiente a la línea, producto, subproducto y tipificación. Para ingresar el lugar de compra, se utiliza una caja de texto y para los comentarios del cliente se utiliza un área de texto que permite la captura de hasta 1000 caracteres.

A través del botón **Guardar** se podrá registrar la información capturada en esta sección, para ello se utilizan los stored procedures `pr_GRConsulta_Producto`, `pr_consultaCAT_cierrellamada`, y las tablas `CAT_PRODUCTO_GRIZZI`, `CIERRE_LLAMADA`. Después de concluir con este proceso la pantalla a la cual nos retornará, será la de búsqueda del cliente para comenzar con el siguiente cliente. (Ver Figura 4.2.2).

En todas las pantallas existe un módulo en la parte inferior nombrado “Cierre de llamada”, este nos da la opción de cerrar la llamada en cualquier momento por cuestiones de que se cortó o se trató de una broma, está conformado por una lista desplegable y un botón llamado **Registrar llamada**, el registro de información se cierra en esta pantalla. Se manejan el stored procedure `pr_InsertaOtra_Llamada` y la tabla `Otra_llamada`.

### 4.3 DESARROLLO DE LOS PROCESOS EN LA BASE DE DATOS (BACK-END)

SQL Server es un sistema de gestión de base de datos relacionales (**RDBMS Relational Database Management System**) diseñado para procesar gran cantidad de información y permite trabajar con aplicaciones comerciales y sitios web.



SQL Server 2000 ofrece el soporte de información para aplicaciones tradicionales cliente/servidor, las cuales están conformadas por una interfaz a través de la cual los clientes acceden a los datos por medio de una LAN.

Hoy en día, se exige un gran porcentaje de distribución de recursos, desconexión a los servidores de datos, los clientes deben estar descentralizados y ser livianos como los navegadores de internet, los cuales accederán a los datos por medio de servicios como el **Internet Information Services (IIS – Servicios de Información de Internet)**.

Existen muchas formas de organizar la información, pero una de las más efectivas está representada por las bases de datos relacionales, las cuales están basadas en la aplicación de la teoría matemática de los conjuntos al problema de la organización de los datos. En una base de datos relacional, los datos están organizados en tablas que representan clases de objetos que tiene importancia para un sistema.

### Instalación de SQL Server (back-end)

Antes de instalar SQL Server 2000 es necesario conocer cuales son los requerimientos mínimos de instalación (Ver Tabla 4.3.1).

Recurso	Requerimiento
Computadora	Intel o compatible
Procesador	Pentium 166 MHz
Monitor	800x600
Dispositivo puntero	Mouse
Tarjeta de red	Opcional (requerido para acceso a los recursos de la red)
CD-ROM	Requerido para la instalación

**Tabla 4.3.1 Requerimientos mínimos para la instalación de SQL Server**

---



Para determinar correctamente el requerimiento de memoria, se puede basar en la Tabla 4.3.2

	<b>Enterprise</b>	<b>Standard</b>	<b>Evaluation</b>	<b>Developer</b>	<b>Personal and desktop engine</b>
Alguna edición de Windows 2000 Server	256 MB (128 MB soportado)	256 MB (128 MB soportado)	256 MB (128 MB soportado)	256 MB (128 MB soportado)	256 MB (128 MB soportado)
Alguna edición de Windows NT 4.0 Server con SP5 ó posterior	128 MB (64 MB soportado)	64 MB	128 MB recomendado (64 MB soportado)	64 MB	32 MB

**Tabla 4.3.2 Memoria**

Para determinar el espacio en disco requerido se debe de tomar la referencia de la Tabla 4.3.3.

<b>Opción de instalación seleccionada</b>	<b>Espacio en disco requerido</b>
Herramientas cliente y servidor	95 – 270 MB dependiendo de las opciones seleccionadas
Instalación típica	250 MB (178 MB para el sistema, mas 72 MB para programas y archivos de datos)
Instalación mínima	110 MB (73 MB para el sistema, mas 37 MB para programas y archivos de datos)
Herramientas administrativas	113 MB (sistema solamente)
Servicios de Análisis	47 MB mínimo, 120 MB típico
English Query	80 MB
Sólo Desktop Engine	44 MB

**Tabla 4.3.3 Espacio en disco duro**



La instalación de SQL Server se puede llevar a cabo de tres formas:

### Instalación local

Algunos de los puntos más relevantes en la instalación de SQL Server son:

- Se deben instalar los service packs del sistema operativo seleccionado.
- Seleccione la opción de instalación de Herramientas clientes y servidor.
- En el cuadro de diálogo Modo de Autenticación, se determina el nivel de seguridad de la instalación de SQL Server. Se puede escoger Modo Windows o Modo Mixto. Si selecciona Modo de autenticación de Windows, todos los permisos de usuario sobre la base de datos se heredarán de la configuración de seguridad de Windows. Si selecciona el Modo Mixto, deberá establecer una contraseña para la cuenta **sa (system administrator - administrador del sistema)** de SQL Server.

### Instalación remota

Para instalar SQL Server en otro equipo que se encuentra en la red, se deben seguir los puntos anteriores y además:

- Tener la información de nombre de cuenta, contraseña, y dominio del equipo donde se instalará SQL Server.
- La ruta de acceso del equipo remoto debe de estar en el formato de convenio de nomenclatura universal (UNC, Universal Naming Conversion), por ejemplo, [\\servidoraplicaciones\c\\$\programas\SQLServer](#).
- Una vez que el proceso de instalación se conecta al equipo remoto, se realiza el proceso de instalación remota.
- Si el proceso de instalación falla, SQL Server crea un archivo llamado Sqlst.log, el cual le ayuda a solucionar el problema.



### Instalación desatendida

Este tipo de instalación permite instalar SQL Server sin tener que estar presente para responder a las peticiones de datos en un archivo, y el programa de instalación los lee automáticamente cuando los necesita. El procedimiento de esta instalación es completamente diferente a las anteriores, se ejecutan algunos de los archivos de procesamiento por lotes que se proporcionan para la ejecución desatendida:

- `Sqlins.bat` Realiza la instalación típica de SQL Server.
- `Sqlcst.bat` Realiza la instalación personalizada de SQL Server.
- `Sqlcli.bat` Instala las utilidades para clientes
- `Sqlrem.bat` Elimina todos los componentes de SQL Server del sistema.

### Creación de una base de datos

SQL Server soporta bases de datos del sistema y del usuario. Las bases de datos del sistema, almacenan información que permite operar y administrar el sistema, mientras que las del usuario almacenan los datos requeridos para las operaciones del cliente.

En términos generales, una base de datos de SQL Server es una colección de objetos que contiene y administra datos. Un archivo de bases de datos no es más que un archivo del sistema operativo. SQL Server divide su trabajo en un archivo para datos y otro para el registro de las transacciones (log).

Una base de datos se puede crear de distintas maneras, una de ellas es utilizando líneas de comando de SQL con la sintaxis siguiente:



```
CREATE DATABASE database_name
[ ON
  { [ PRIMARY ] ( [ NAME = logical_file_name ,
    FILENAME = 'os_file_name'
    [ , SIZE = size ]
    [ , MAXSIZE = { max_size | UNLIMITED } ]
    [ , FILEGROWTH = growth_increment ] )
  } [ ,...n ]
]
[ LOG ON
  [ NAME = logical_file_name ,
    FILENAME = 'os_file_name'
    [ , SIZE = size ]
    [ , MAXSIZE = { max_size | UNLIMITED } ]
    [ , FILEGROWTH = growth_increment ] )
  } [ ,...n ]
]
[ COLLATE collation_name ]
```

Cuando se crea una base de datos se deben especificar los siguientes parámetros:

**PRIMARY** - Este parámetro especifica los archivos en el grupo de archivos primario. Este grupo contendrá todas las tablas de sistema de la base de datos. Este archivo es el punto de acceso a la base de datos y al resto de los archivos de la base de datos.

**FILENAME** – Especifica el nombre y la ruta del archivo de sistema operativo.

**SIZE** – Este parámetro especifica el tamaño del archivo de datos o del log.

**MAXSIZE** – Este parámetro especifica el tamaño máximo hasta el cual puede crecer el archivo.

**FILEGROWTH** - Este parámetro especifica el incremento de crecimiento del archivo.

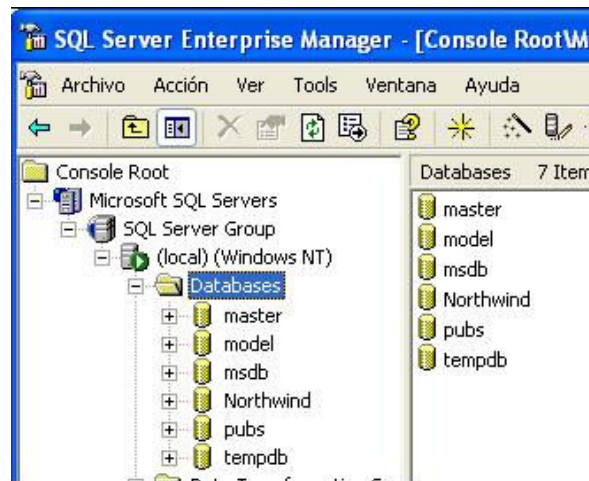
**COLLATION** – Aquí se especifican los parámetros de la base de datos, como el lenguaje o el alfabeto.

A continuación se muestra un ejemplo para la creación de la base de datos Grizzi



```
USE master
GO
CREATE DATABASE GRIZZI
ON
( NAME = Grizzi_dat,
  FILENAME = 'c:\program files\microsoft sql server\mssql\data\Grizzidat.mdf',
  SIZE = 10,
  MAXSIZE = 50,
  FILEGROWTH = 5 )
LOG ON
( NAME = 'Grizzi_log',
  FILENAME = 'c:\program files\microsoft sql server\mssql\data\Grizzilog.ldf',
  SIZE = 5MB,
  MAXSIZE = 25MB,
  FILEGROWTH = 5MB )
GO
```

Otra manera de crear una base de datos es utilizando el **wizard (asistente de creación de bases de datos)** del Administrador Corporativo y seleccionando la carpeta de bases de datos. (Ver Figura 4.3.4)



**Figura 4.3.4 Administrador corporativo**

En el menú **Herramientas** seleccione la opción **Asistentes**, y después la opción Asistente para la creación de bases de datos (ver Figura 4.3.5).



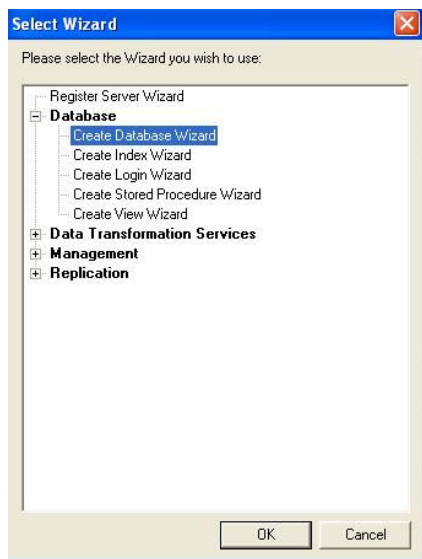


Figura 4.3.5 Asistente para la creación de bases de datos

Se presentará una pantalla de bienvenida al asistente, de click en siguiente. (Ver Figura 4.3.6)

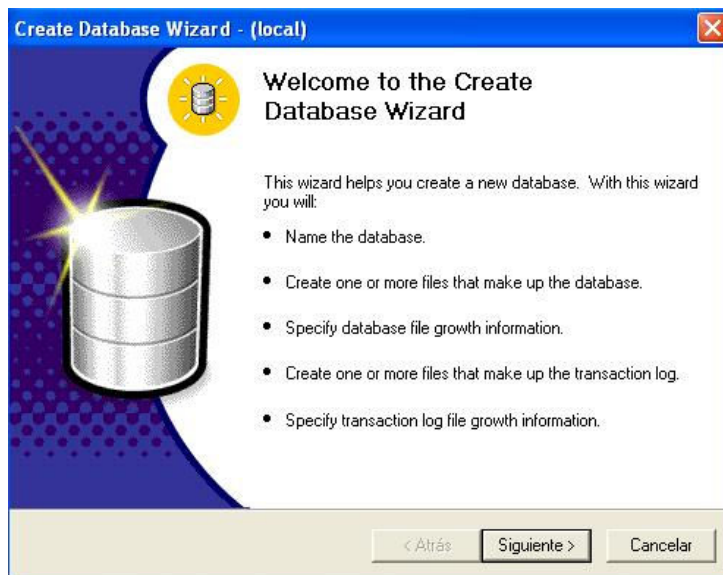


Figura 4.3.6 Asistente para crear bases de datos

La siguiente pantalla le permite especificar el nombre de la base de datos y las carpetas donde se almacenarán los archivos de datos y el log. (Ver Figura 4.3.7)

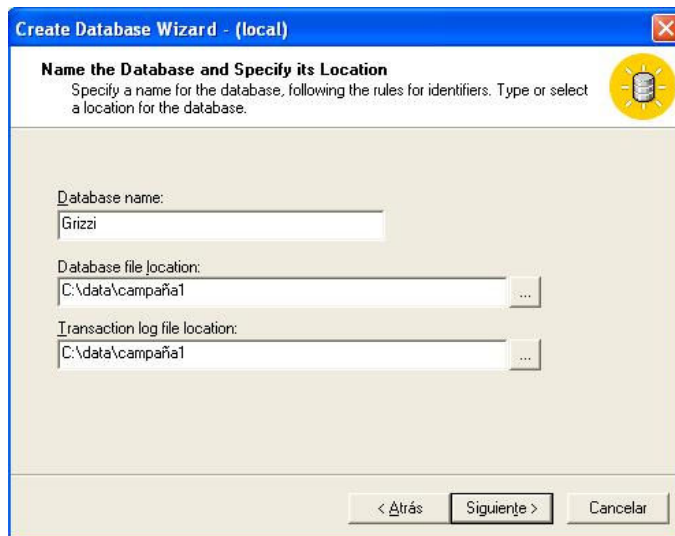


Figura 4.3.7 Nombre de la base de datos, ruta y log de la base de datos

Después de dar click en **Siguiente**, aparecerá una pantalla donde se debe especificar si se desea emplear más de un archivo de datos e indicar el tamaño de cada uno. (Ver Figura 4.3.8)

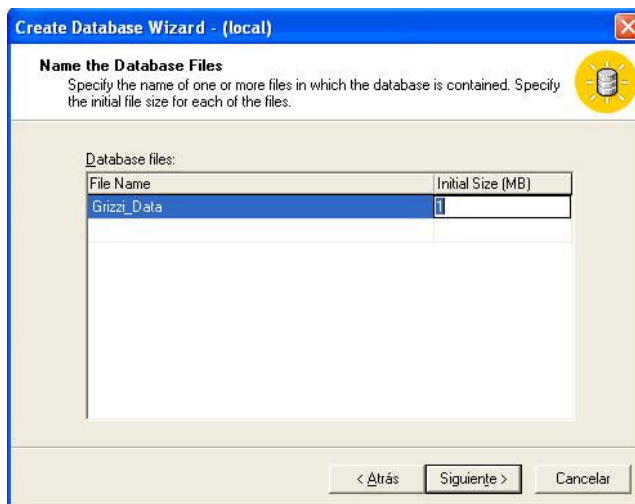


Figura 4.3.8 Archivos de datos adicionales

Oprima el botón **Siguiente**, aparecerán las opciones para personalizar el crecimiento del archivo de datos. (Ver Figura 4.3.9)

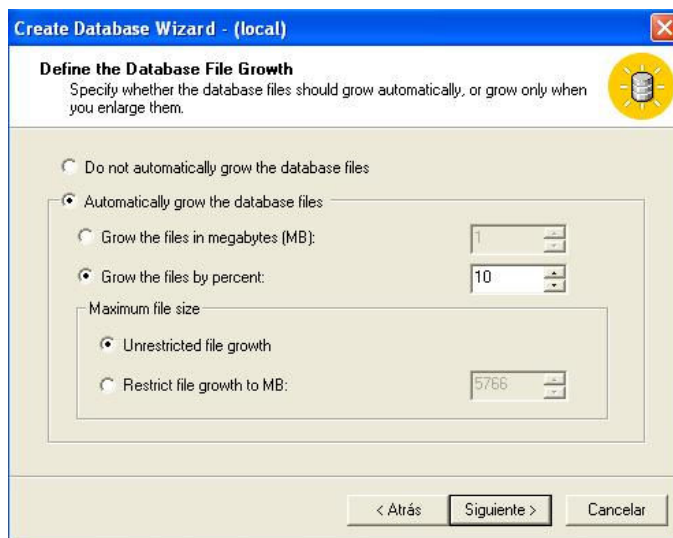


Figura 4.3.9 Personalizando el crecimiento automático del archivo de datos

Continuando con el proceso, al oprimir **Siguiente**, se debe de especificar el nombre para el archivo log y su tamaño inicial (Ver Figura 4.3.10).

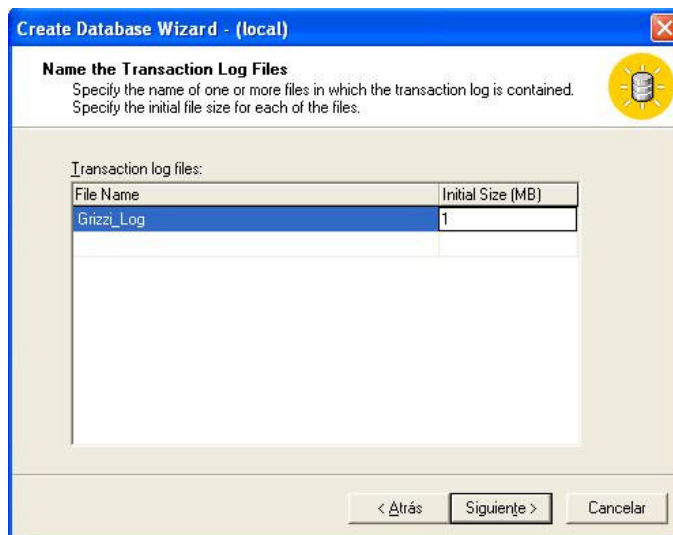


Figura 4.3.10 Especificación del archivo log

Al igual que en el archivo de datos, después de dar click en **Siguiente** también se podrá establecer el crecimiento del archivo log. (Ver Figura 4.3.11)



Figura 4.3.11 Parámetros del archivo de transacciones

Seleccione el botón **Siguiete** y aparecerá una pantalla en la cual se muestran las características con las que se creará la base de datos (ver Figura 4.3.12).

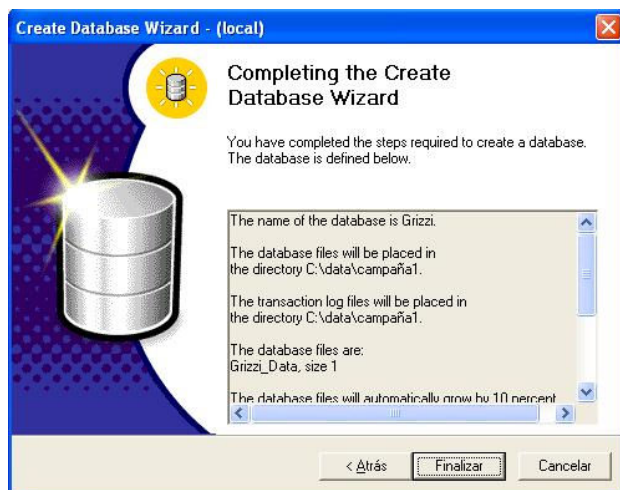
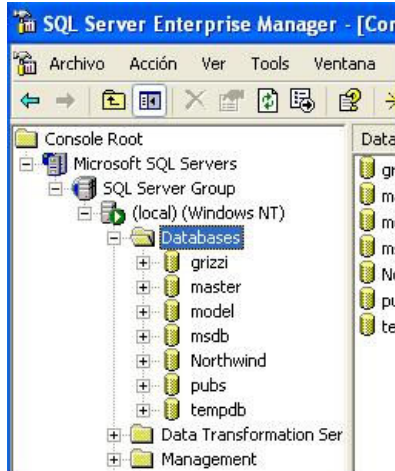


Figura 4.3.12 Finalización del proceso

Al seleccionar **Finalizar**, se creará la base de datos y se mostrará un mensaje en donde se pregunta si se desea crear un plan de mantenimiento para la nueva base, al cual responderemos que no. Una vez finalizado este proceso podremos



visualizar la nueva base de datos desde el Administrador Corporativo. (Ver Figura 4.3.13)



**Figura 4.3.13** Árbol de bases de datos

### Creación de una tabla

Desde el Administrador Corporativo, se pueden observar todos los objetos que contiene una base de datos por ejemplo: tablas, vistas, procedimientos almacenados. Con respecto a las tablas, podemos mencionar que son colecciones ordenadas de filas y columnas que almacenan información de objetos simples. Son una unidad fundamental de almacenamiento en una base de datos relacional.

Cada tipo de entidad se representa por una tabla y dentro de ella, cada instancia de esa unidad se representa por una fila de la tabla. Las tablas se crean y se mantienen utilizando el diseñador de tablas del Administrador corporativo.

Se pueden crear tablas de distintas maneras, una de ellas es utilizando líneas de comando de SQL con la sintaxis siguiente:



```
CREATE TABLE [ database_name.[ owner ] . | owner. ] table_name
  ( { < column_definition >
    | < table_constraint > } [ ,...n ] )
```

```
< column_definition > ::= { column_name data_type }
  [ [ DEFAULT constant_expression ]
  [ < column_constraint > ] [ ...n ]
```

```
< column_constraint > ::= [ CONSTRAINT constraint_name ]
  | [ { PRIMARY KEY | UNIQUE }
    [ CLUSTERED | NONCLUSTERED ] ]
  | [ [ FOREIGN KEY ]
    REFERENCES ref_table [ ( ref_column ) ]
    [ ON DELETE { CASCADE | NO ACTION } ]
    [ ON UPDATE { CASCADE | NO ACTION } ] ]
  | CHECK ( logical_expression ) }
```

```
< table_constraint > ::= [ CONSTRAINT constraint_name ]
  { [ { PRIMARY KEY | UNIQUE }
    [ CLUSTERED | NONCLUSTERED ]
    { ( column [ ASC | DESC ] [ ,...n ] ) } ]
  | FOREIGN KEY
    [ ( column [ ,...n ] ) ]
    REFERENCES ref_table [ ( ref_column [ ,...n ] ) ]
    [ ON DELETE { CASCADE | NO ACTION } ]
    [ ON UPDATE { CASCADE | NO ACTION } ]
  | CHECK ( search_conditions ) }
```

Cuando se crea una tabla, se debe especificar el nombre de la tabla, el nombre de las columnas y el tipo de dato. Los nombres de las columnas deben ser únicos para una tabla específica, pero se puede utilizar el mismo nombre de columna en diferentes tablas de una misma base de datos.

El uso de **constrains (restricciones)** es el mejor método para mantener la integridad de la información. Con esto se asegura que sólo datos válidos podrán ser agregados a las tablas y las relaciones entre ellas permanecerán.



Se tiene tres tipos de integridad de datos:

- Dominio (columnas) - especifica el conjunto de datos que serán válidos para una columna y determina si son permitidos valores nulos.
- Entidad (renglones) – requiere que todos los renglones de una tabla tengan un identificador único, conocido como llave primaria.
- Referencial (entre tablas) – Asegura que las relaciones entre llaves primarias y foráneas se mantengan siempre

A continuación se presenta un ejemplo de la creación de una tabla:

```
CREATE TABLE [dbo].[CIERRE_LLAMADA] (  
    [iCierreID] [int] NOT NULL ,  
    [cCierreDescripcion] [varchar] (30) ,  
    [bCierreStatus] [bit] NOT NULL  
) ON [PRIMARY]
```

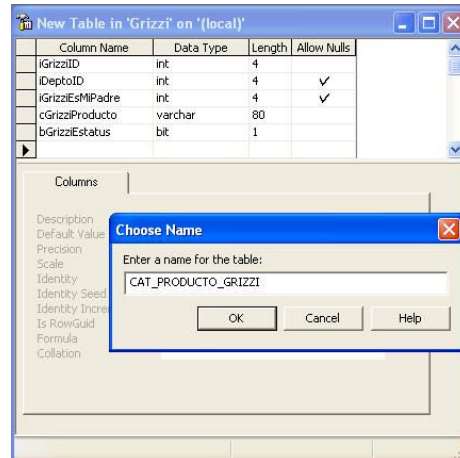
Otra manera de crear una tabla es utilizando el wizard, para crear una nueva tabla dentro de la base de datos deberá seleccionar la opción **Tablas** y después dar un click derecho sobre la opción **Nueva tabla**. (Ver Figura 4.3.14).





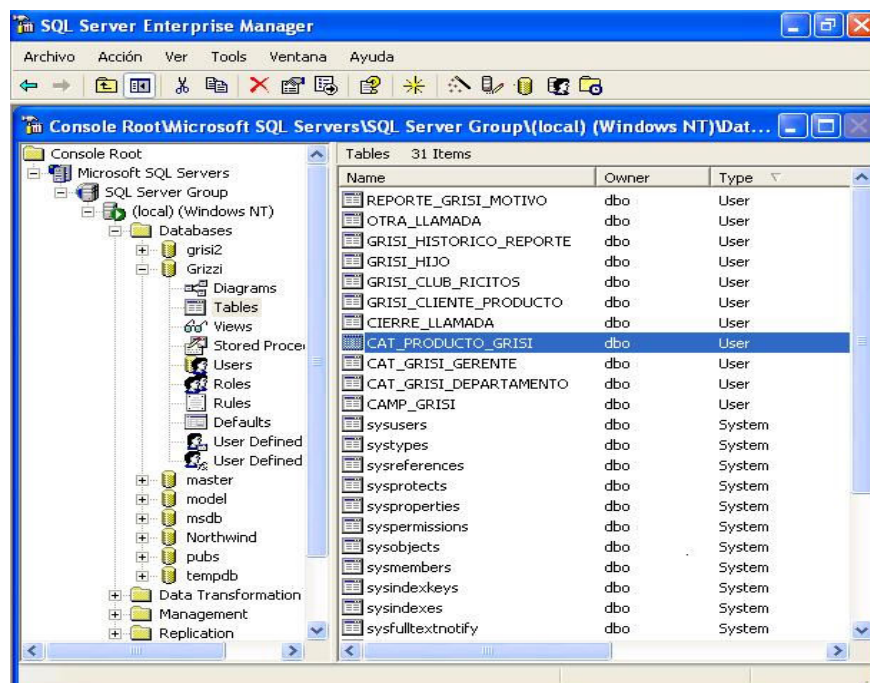


Al finalizar la captura de las columnas, dé click en el botón **Guardar**, escriba el nombre de la nueva tabla y seleccione **OK** (Ver Figura 4.3.16).



**Figura 4.3.16** Nombrando la nueva tabla

Se podrá visualizar la nueva tabla en el panel de la derecha del Administrador corporativo, desde donde se podrán agregar o eliminar columnas en todas las tablas. (Ver Figura 4.3.17)

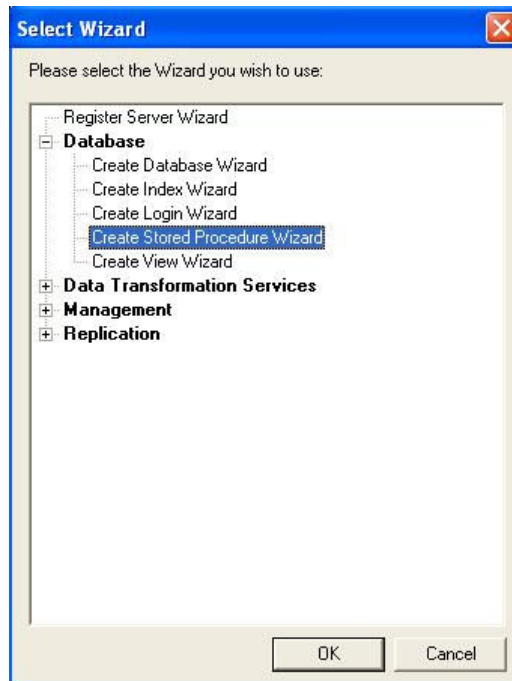


**Figura 4.3.17** Árbol de tablas



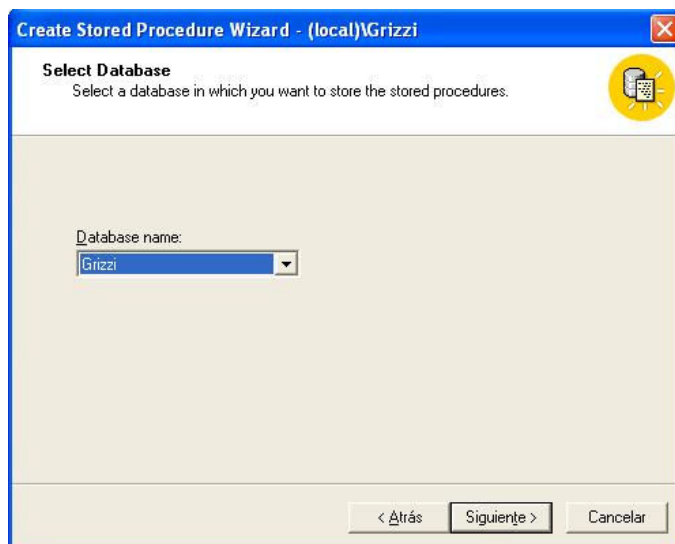
### Creación de un procedimiento almacenado (stored procedure)

Para crear un stored procedure en SQL Server desde el Administrador corporativo, seleccione el menú Herramientas, de click en la opción Asistentes y seleccione Create Stored Procedure Wizard (ver Figura 4.3.18).



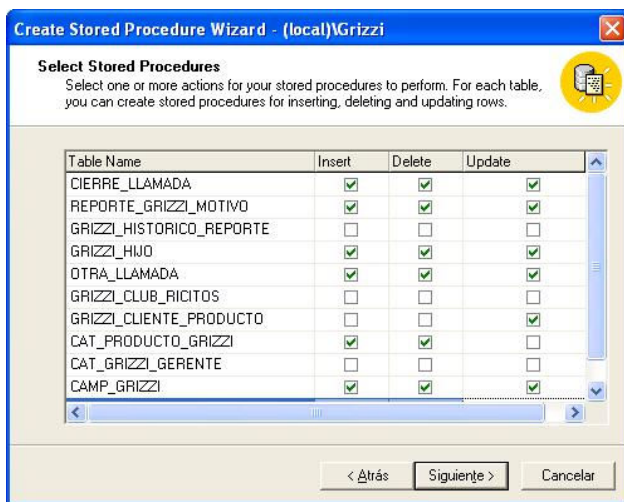
**Figura 4.3.18 Asistente para crear stored procedures**

Enseguida, se desplegará una pantalla de bienvenida al asistente, de click en **Siguiente** y seleccione la base de datos que utilizará. (Ver Figura 4.3.19)



**Figura 4.3.19 Elección de base de datos**

Oprima el botón **Siguiente**, se desplegará una pantalla en la que deberá seleccionar las acciones que se realizarán sobre las tablas (insert, delete o update), las cuales se almacenarán en el stored procedure. (Ver Figura 4.3.20).



**Figura 4.3.20 Selección de acciones para el stored procedure**

De click en **Siguiente**, se mostrará una pantalla en la que se listarán todas las acciones que se ejecutarán en el stored procedure. (Ver Figura 4.3.21)



Figura 4.3.21 Finalización del proceso

Por último, de click en **Finalizar**, se desplegará un mensaje indicando que el stored procedure ha sido creado y se visualizará en el panel de la derecha del Administrador corporativo. (Ver Figura 4.3.22).

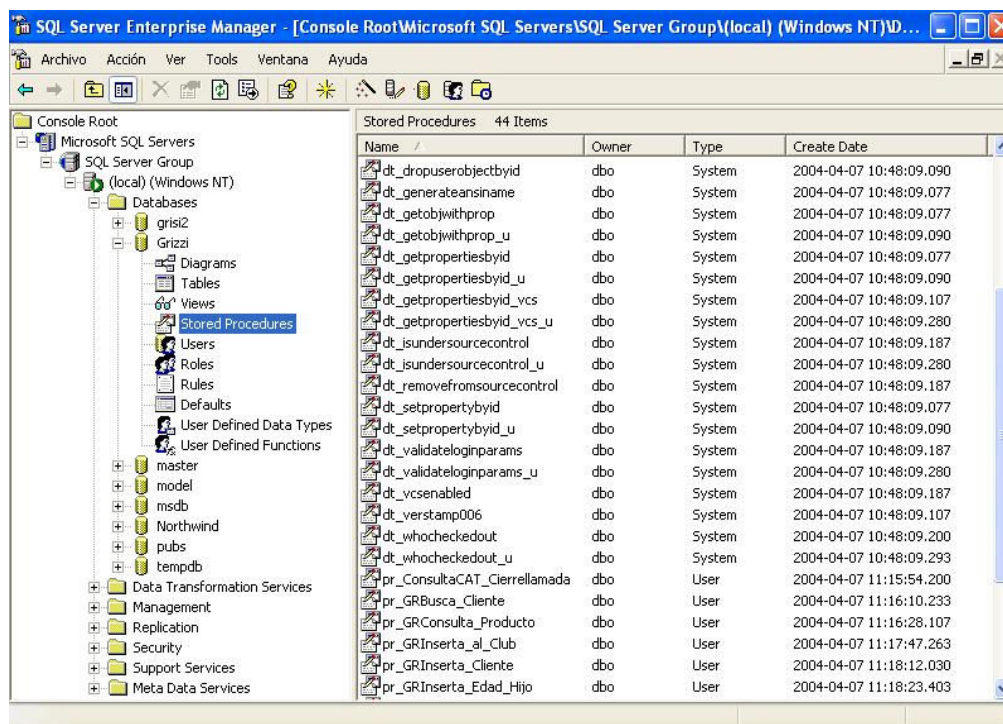


Figura 4.3.22 Árbol de stored procedure



Para crear un procedimiento almacenado desde la línea de comandos, utilizamos la siguiente sintaxis:

```
CREATE PROC [ EDURE ] procedure_name [ ; number ]  
    [ { @parameter data_type }  
      [ VARYING ] [ = default ] [ OUTPUT ]  
    ] [ ,...n ]  
[ WITH  
    { RECOMPILE | ENCRYPTION | RECOMPILE , ENCRYPTION } ]  
[ FOR REPLICATION ]  
AS sql_statement [ ...n ]
```

A continuación se presenta un ejemplo para la creación de un stored procedure:

```
CREATE PROCEDURE pr_AccesoSistema  
@iID int = NULL,  
@cPassword char(8) = NULL  
AS  
IF NOT EXISTS (SELECT iID FROM GRIZZI_EMPLEADOS WHERE iID = @iID  
AND cContraseña = @cPassword)  
BEGIN  
    SELECT 'Error' = -1, 'Descripcion'='USUARIO O PASSWORD INVALIDO'  
    RETURN(-1)  
END  
SELECT *  
FROM GRIZZI_EMPLEADOS  
WHERE iID = @iID  
AND cContraseña = @cPassword
```



## 4.4 GENERACIÓN DE REPORTE

Una de las partes fundamentales del sistema es la generación de los reportes, que permiten analizar entre otras cosas:

- El comportamiento de la venta de los productos.
- Los establecimientos donde se vende más un cierto tipo de producto.
- El nivel de productividad en lo que se refiere a la atención de llamadas.
- Tener una base de datos más representativa para el fin del posicionamiento de los productos en el mercado.

En la Figura 4.4.1, podemos observar el porcentaje de las edades de los niños que consumen algunos de los productos de cierta marca, con estos datos el área de mercadotecnia puede realizar las campañas necesarias para la venta de nuevos productos, teniendo en cuenta el tipo de mercado al que se dirige.

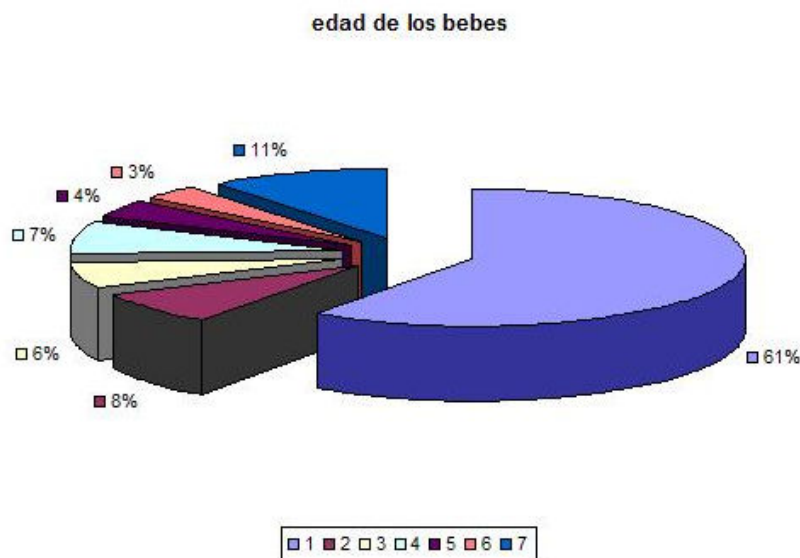


Figura 4.4.1 Gráfica de edades



En la Figura 4.4.2 observamos los estados que presentan un mayor consumo de productos de una marca, lo cual indica hacia donde se dirige el mercado, y con esto poder tomar decisiones sobre que productos introducir en las diferentes comunidades, o hacer promociones de los productos que no están siendo adquiridos.

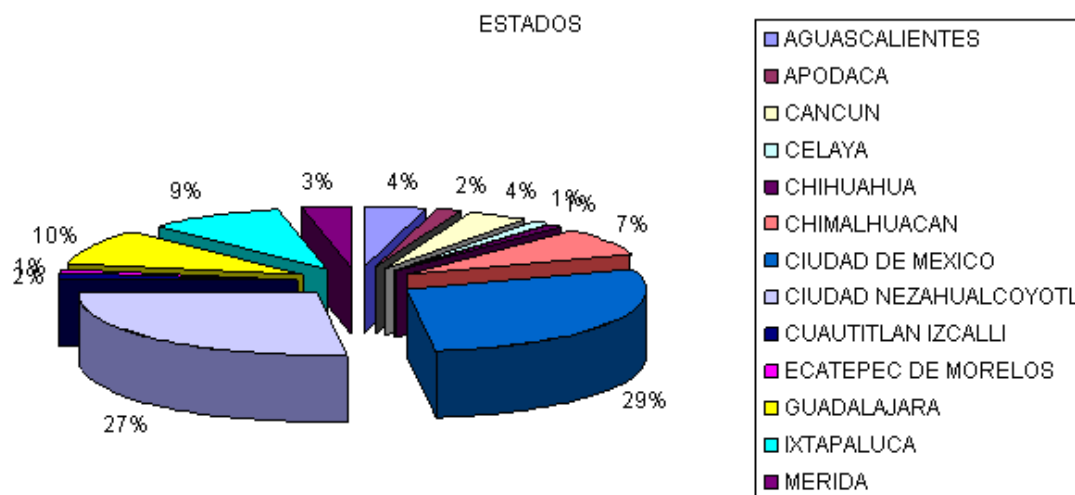


Figura 4.4.2 Gráfica de consumo de productos por comunidades

En la Figura 4.4.3 se puede observar el porcentaje de las personas a las que se les preguntó si deseaban pertenecer a algún club o participar en algún sorteo de los productos que la marca esta realizando.



Figura 4.4.3 Gráfica de inscripción al club

Las gráficas de las Figuras 4.4.1 a 4.4.3 son una muestra de lo que el sistema puede realizar para la generación de reportes.

## 4.5 INTEGRACIÓN Y PRUEBAS DEL SISTEMA

### Tipos de pruebas

En la construcción de todo sistema, es indispensable realizar pruebas para estar seguros de que todo funciona como se espera y, en caso de no ser así, llevar a cabo la corrección y depuración de los elementos que presenten problemas para mejorar la calidad del sistema.

En el proceso de elaboración de software, como en cualquier otro proceso humano que requiera de atención, precisión, comunicación, comprensión y creación en un período determinado, inevitablemente se tienen defectos, los cuales deben ser detectados y corregidos para la entrega de un producto determinado libre de fallas (dentro de lo posible).





Comúnmente se tiene la idea de que la depuración de un sistema comienza una vez que éste ha sido terminado, sin embargo, esto no es del todo cierto. La depuración debe realizarse una vez que cada uno de los módulos que conformarán al sistema se encuentre terminado y, de la misma forma, cuando dichos módulos se integran. Para esta depuración, se realizan distintos tipos de pruebas, entre las que podemos mencionar las principales:

- **Pruebas de caja negra.** Son aquellas que se enfocan directamente en el exterior del módulo, sin importar el código. Este tipo de pruebas es aplicable a los módulos que trabajan en forma transparente para el usuario.

Para realizar estas pruebas existe una técnica algebraica llamada “clases de equivalencia”, consisten en tratar a todas las posibles entradas y parámetros como un modelo algebraico, y utilizar las clases de este modelo para probar un amplio rango de posibilidades. Para la generación de estas clases no se puede armar un modelo, pero se pueden seguir las siguientes pautas como guía utilizable para la creación de cada clase.

Cuando una entrada es booleana, existen solo dos clases, verdadero o falso.

Para una entrada que está comprendida dentro de un rango, existen tres clases, por debajo, dentro y por encima del rango.

Utilizando esta técnica se pueden generar las distintas clases aplicables al módulo en cuestión, luego se procede a ingresar al módulo algún valor de cada clase.

- **Pruebas de caja blanca.** Son mucho más amplias, normalmente se denominan pruebas de cobertura o pruebas de caja transparente; al total de



pruebas de caja blanca se le llama cobertura. La cobertura es un número porcentual que indica cuanto código del programa se ha probado. Básicamente, la idea de pruebas de cobertura consiste en diseñar un plan en el que se vaya ejecutando sistemáticamente el código hasta que haya corrido todo o la gran mayoría de éste.

Para los módulos que no poseen condiciones, basta con ejecutar una vez el programa para asegurar una cobertura total. Es importante que el diseño de cobertura sea eficiente y lo menos redundante posible, por ejemplo, en el siguiente código:

```
If Variable_booleana then  
... call Módulo_x  
End if
```

Como no hay un else, a simple vista con ejecutar una vez con éxito la condición bastaría, en términos de cobertura es así, pero entendiendo que el “Módulo\_x” podría modificar variables o valores que afecten a la ejecución del resto del código habría que ejecutar dos veces la condición, una satisfaciendo y la otra no.

En el siguiente ejemplo:

```
If Variable_booleana1 or Variable_booleana2 then  
..call Módulo_x  
End if
```

A simple vista y considerando que ambas variables pueden tomar dos valores se precisarían cuatro pruebas para realizar la cobertura, pero esto no es así, sólo son necesarias dos pruebas en el caso de que el “Módulo\_x” pueda inferir de alguna manera en el programa o una sola satisfaciendo la

---



condición si el “Módulo\_x” no alterará de ninguna manera el resto de los procedimientos y condiciones a ejecutarse. Probado las cuatro posibilidades se estaría probando la funcionalidad de la sentencia if en sí, lo cual ya ha sido probado por el desarrollador del lenguaje de programación.

Con respecto a la cobertura de ciclos el tema es más delicado, a simple vista un ciclo no es más que un salto condicional que se repite hasta que se cumpla o deje de cumplirse una o más condiciones, en teoría esto es simple, pero en la práctica son una fuente inagotable de versátiles errores, que en su gran mayoría suelen ser catastróficos. En primer lugar, la cantidad de veces que se ejecute un ciclo debe ser precisa, y todos los programadores tienen claro que no es difícil equivocarse y programar un ciclo que se ejecute una vez de más o una vez de menos, siempre que esto suceda los resultados serán indeseables, y muchas veces cuando se trate de manejo de datos complicados de calcular no será fácil advertir el error, el cual será costoso cuando se trate de valores que se utilizan para tomar decisiones a nivel empresarial o involucren vidas humanas.

Para realizar la cobertura total de un ciclo se necesitan tres pruebas, cero ejecuciones, una ejecución y más de una ejecución.

Los ciclos “for”, parecerían ser más sencillos, ya que la cantidad de ejecuciones es definida por su cabecera y controlada por el compilador, con una ejecución bastaría para una cobertura total, siempre y cuando no contengan código que altere el valor de la variable de control o comandos de salida (Exit), en este caso requiere un examen un poco más detallado, ya que el ciclo deja de ser responsabilidad del lenguaje compilador y pasa a ser del programador.



En pocas palabras, es muy importante diseñar lo más precisamente posible las pruebas de cobertura, para que quede en lo posible la mayor parte del código probado con la mínima cantidad de pruebas realizadas.

Hay que tener en cuenta dos puntos importantes, en primer lugar las pruebas de caja blanca no reemplazan, solo complementan a las de caja negra y de aceptación, y en segundo lugar, las pruebas de cobertura deben ser realizadas una vez terminado el software y no deben ser confundidas con las pruebas informales que realiza el programador en momentos de desarrollo, dado que si bien, éstas van cubriendo distintos fragmentos de cada módulo, nunca son eficaces por no tener un diseño apropiado.

- **Pruebas de integración.** Se llevan a cabo durante la construcción del sistema, involucran a un número creciente de módulos y terminan probando el sistema como conjunto. Estas pruebas se pueden plantear desde un punto de vista estructural o funcional. Las pruebas estructurales de integración son similares a las pruebas de caja blanca, pero trabajan a un nivel conceptual superior. En lugar de referirnos a sentencias del lenguaje, nos referimos a llamadas entre módulos. Se trata pues, de identificar todos los posibles esquemas de llamadas y ejercitarlos para lograr una buena cobertura de segmentos o de ramas.

Las pruebas funcionales de integración son similares a las pruebas de caja negra. Aquí trataremos de encontrar fallos en la respuesta de un módulo cuando su operación depende de los servicios prestados por otro(s) módulo(s). Según nos vamos acercando a la totalidad del sistema, estas pruebas se van basando más y más en la especificación de requisitos del usuario.



Las pruebas finales de integración cubren todo el sistema y pretenden cubrir plenamente la especificación de requisitos del usuario. Además, a estas alturas ya suele estar disponible el manual de usuario, que también utiliza para realizar pruebas hasta lograr una cobertura aceptable. En todas estas pruebas funcionales se siguen utilizando técnicas de partición en clases de equivalencia y análisis de casos límite (frontera).

- **Pruebas alfa.** Consisten en invitar al cliente a que venga al entorno de desarrollo a probar el sistema. Se trabaja en un entorno controlado y el cliente siempre tiene un experto a mano para ayudarlo a utilizar el sistema y analizar los resultados.
  
- **Pruebas beta.** Estas pruebas se ejecutan después de las pruebas alfa, y se desarrollan en el entorno del cliente, en un entorno fuera de control del desarrollador. Aquí el cliente queda a solas con el producto y trata de encontrarle fallas (reales o hipotéticas) de las que informa al desarrollador. Las pruebas alfa y beta son habituales en productos que se van a vender a gran cantidad de clientes. Algunos de los potenciales compradores se prestan a este tipo de pruebas para ir entrenando a su personal con tiempo, o a cambio de alguna ventaja económica (mejor precio sobre el producto final, derecho a mantenimiento gratuito por tiempo a nuevas versiones del producto, etc.). Estas prácticas resultan muy eficaces.
  
- **Pruebas de regresión.** Consisten en volver a ejecutar un subconjunto de pruebas que se han llevado a cabo anteriormente para asegurarse que los cambios no han propagado efectos colaterales no deseados.
  
- **Pruebas de validación.** Proporcionan una seguridad final de que el software satisface todos los requisitos funcionales de comportamiento y



rendimiento. Durante la validación se usan exclusivamente pruebas de caja negra.

- **Pruebas de resistencia (stress).** Estas pruebas se ejecutan en el sistema de forma que demande recursos en cantidad, frecuencia o volúmenes anormales, es decir, están diseñadas para enfrentar a los programas con situaciones poco comunes.
- **Pruebas de robustez.** Comúnmente denominadas “robustness test” son las encargadas de verificar la capacidad del sistema para soportar entradas incorrectas.
- **Pruebas de módulo.** Un módulo está integrado de funciones que pueden cooperar entre si, después de haber probado cada función individual, es necesario probar la operación de estas funciones cuando componen un módulo. Debe ser posible probar un módulo como una entidad aislada, sin la presencia de otros módulos del sistema.

### Pruebas realizadas al sistema

Se realizaron pruebas de Caja Blanca y Caja Negra al sistema desarrollado, con los siguientes resultados:

#### Caja blanca

Se probará el siguiente código involucrado en la funcionalidad de ingreso de usuarios al sistema, el cuál está constituido por los siguientes módulos:

- Index.asp
- Valida.js
- Funciones.asp



Los lenguajes que fueron utilizados son HTML, VBScript y ASP. El sistema ha sido desarrollado para funcionar en una intranet.

### Index.asp

```
<!-- #include file="tools\funciones.asp"-->
<%
if request("s")<>"" then
    session.Abandon
    response.redirect "index.asp"
end if

if session("agente")<>"" then response.redirect "busqueda.asp"
call Header(0,0)

if request.form()<>"" then
    call Validacion("pr_AccesoSistema",3,request.form("usuario"),request.form("password"),"busqueda.asp")
else
    if request("msg")<>"" then%>
        <script lenguaje="javascript">alert('<%=request("msg")%>');</script>
    <%end if%>

<form action="index.asp" name="logueo" method="post" onSubmit="return v_login();">
<table frame="0" cellspacing="5" cellpadding="0" align="center" valign="middle">
<tr><td colspan=2 align=center><b>Inicio de sesión.</b></td></tr>
<tr><td align="right">Usuario: </td>
<td><input name="usuario" type="TEXT" size="20" maxlength=20 onkeypress=
"Validkey('usuario');"></td></tr>
<tr><td align="right">Password: </td>
<td><input name="password" TYPE="PASSWORD" size="20" maxlength=20
onkeypress="Validkey('usuario');"></td></tr>
<tr><td colspan="2" align=center><input type="submit" name="submit" value="Entrar"></td></tr>
</table>
</form>
<%end if
call Footer()%>
```



En el código anterior se puede identificar lo siguiente:

- La etiqueta FORM tiene el atributo METHOD=POST, lo cual es adecuado cuando se envía la información a través del formulario
- Se permite la captura de texto para los campos usuario y password, con una longitud máxima de 20 caracteres alfanuméricos
- El botón “Entrar” dirige la captura al módulo v\_login

### Valida.js

```
function v_login(){
    if (Trim(document.logueo.usuario.value).length == ""){
        alert("Debe proporcionar el usuario");
        document.logueo.usuario.focus();
        return false;
    }
    if (Trim(document.logueo.password.value).length == ""){
        alert("Debe proporcionar el password");
        document.logueo.password.focus();
        return false;
    }
    return true;
}
```

En el código anterior, se puede identificar lo siguiente:

- En caso de que el campo usuario no tengan información, mandará el mensaje de error “Debe proporcionar el usuario”, regresando a la misma pantalla, haciendo esta validación del lado del cliente.
- En caso de que el campo password no tengan información alguna, mandará el mensaje de error “Debe proporcionar el password”, regresando a la misma pantalla, haciendo esta validación del lado del cliente, es importante





esta parte, ya que al realizar este tipo de validación, se evita generar un envío de paquetes innecesario a través de la red.

### Funciones.asp

```
sub Validacion(sqlstring,idconexion,Usuario>Password,Accion)
Dim validpwd
call OpenConexion(idconexion)
sqlAdd = "exec "&sqlstring&" ""& Usuario &" ""& Password &""
rs.Open sqladd, conn
If rs(0)="0" then
    session("nomagente")=rs(3)
    Session("status") = "login"
    Session("agente") = usuario
    call CloseConexion()
    response.redirect Accion
Else
    msg1=rs(1)
    call CloseConexion()%>
    <script language="javascript">alert("<%=msg1%>");history.back();</script>
<%end if
end sub

sub OpenConexion(accion)
Set Conn = Server.CreateObject("ADODB.Connection")
select case accion
case 1:
    SERVIDOR = "x"
    BASE = "y"
    USUARIO = "u"
    PASSWORD = "c"
case 2:
    SERVIDOR = "s"
    BASE = "y2"
    USUARIO = "u2"
    PASSWORD = "c2"
```



```
end select
strConn = "Provider=SQLOLEDB; Data Source="&SERVIDOR&; Initial Catalog="&BASE&;
        User Id="&USUARIO&; Password="&PASSWORD&;"
Conn.Open strConn
Set rs = Server.CreateObject ("ADODB.Recordset")
end sub

sub CloseConexion()
    if not rs.eof then
        rs.close
    end if
    set rs=nothing
    Conn.close
    set Conn=nothing
end sub
```

En el código anterior identificamos lo siguiente

- La función Validación realiza una conexión a las Bases de Datos donde se encuentra la información de los usuarios y, en caso de ser aceptados, la base donde se encuentran los códigos postales, que servirán para cargar o capturar la información del cliente, en caso de que el usuario y la contraseña sean válidos, abre una sesión de ASP y carga la página `busqueda.asp`. En caso contrario, extrae de la base de datos el mensaje de error y lo despliega en la página `index.asp`.
- La función `OpenConexion` realiza la conexión a la base de datos utilizada en la función Validación.
- La función `CloseConexion` se encarga de cerrar la conexión a la base de datos una vez que se han hecho las consultas necesarias.



### Caja negra

Las siguientes interfaces de usuario están involucradas en la funcionalidad del ingreso del usuario al sistema. La funcionalidad esperada es si el usuario captura información válida en la pantalla de acceso, se permite el ingreso a la página correspondiente, en caso contrario, indica las razones por las que no se puede ingresar.

La Figura 4.5.1 muestra la pantalla de acceso donde se capturan los datos del usuario que quiere ingresar. Si el usuario no captura datos en el campo 'usuario' se muestra la Figura 4.5.2, y si no captura el campo 'password' se muestra en la Figura 4.5.3.

La imagen muestra una interfaz de usuario para el acceso al sistema. En la parte superior derecha hay un botón que dice "CERRAR". El texto principal indica: "Por favor ingrese la siguiente información para acceder a la aplicación." Debajo de esto hay dos campos de entrada: "Usuario:" y "Password:". Debajo de los campos hay un botón que dice "ENTRAR".

**Figura 4.5.1 Pantalla de acceso al sistema**



Figura 4.5.2 Pantalla con campo 'usuario' vacío



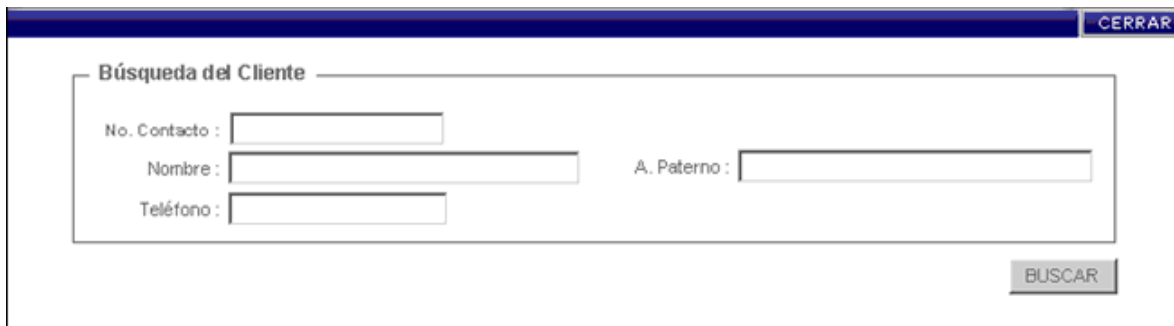
Figura 4.5.3 Pantalla con campo 'password' vacío

Cuando el usuario captura los datos usuario/password con datos que no son válidos, la pantalla de la Figura 4.5.4 se presenta.



**Figura 4.5.4 Pantalla de usuario/contraseña incorrecta**

Cuando el usuario captura los campos usuario/password con datos válidos, se presenta la pantalla de la Figura 4.5.5 que corresponde a la siguiente pantalla en el proceso.



**Figura 4.5.5 Pantalla de usuario/contraseña válidos**



### 4.6 OPTIMIZACIÓN DE SQL SERVER (SQL SERVER TUNING)

Para realizar la afinación del sistema, previamente se tienen que realizar algunas pruebas para ubicar donde se encuentran los conflictos que hacen que nuestro sistema no responda como se tiene planeado.

La meta a alcanzar para poner a punto el sistema, es establecer una base de desempeño que pueda hacer posible ejercer mayor control sobre el uso del sistema y estar preparado para cambios repentinos a ese uso. La base de desempeño consiste en definir rangos aceptables de funcionamiento de cada uno de los recursos del sistema y calificar umbrales máximos de acción, los cuales no deben ser alcanzados.

Cuando los umbrales definidos para cada uno de los recursos son alcanzados o superados, debemos tener claramente definidas las acciones a realizar para normalizar su funcionamiento.

La finalidad que se persigue con la afinación del desempeño del sistema, es:

- Maximizar el flujo de la información.
- Mantener un tiempo de respuesta aceptable
- Utilizar eficientemente los recursos de éste.

Se puede observar que la principal meta de la afinación del sistema es la de maximizar los procesos realizados por el mismo por minuto, mientras se mantiene un tiempo de respuesta aceptable. Así, la afinación nos permitirá hacer uso de los recursos de la mejor manera posible.



Para la optimización de SQL Server es necesario ajustar la configuración y opciones. Estas observaciones se realizan cuando el servidor se encuentra en condiciones críticas de trabajo, debido a que en esta circunstancia el servidor está con la carga de trabajo más pesada. Las siguientes recomendaciones para la afinación del sistema son las que tienen mayor impacto en el trabajo del servidor.

Las opciones explicadas aquí se definen mediante el procedimiento almacenado SP\_CONFIGURE o con la interfaz gráfica del Administrador corporativo en la configuración del servidor.

### **Backup threads**

Esta opción de configuración muestra el número de hilos de servicio de NT que se asignarán a las copias de seguridad por tiras y también a las operaciones de carga. Deben de incrementarse para mejorar los tiempos de copia de seguridad y de carga.

### **Data base size**

Esta opción controla el número de megabytes por omisión que se reservan para una nueva base de datos cuando ésta se crea. Si la mayoría de las bases de datos que se vayan a crear en un servidor dado, son mayores que 2 MB, sería recomendable modificar este valor. Si la base de datos modelo se extiende hasta ser mayor de 2 MB, será necesario ajustar este valor.

### **Default SortOrder Id**

Cuando se selecciona esta opción, se controla el orden de clasificación que utilizará el servidor, la forma en que SQL Server ordena los datos y los proporciona al cliente. El valor por omisión es Dictionary, Case Insensitive.



### **Medía retention**

Esta opción de configuración controla el número de días que debe de ser retenida una copia de seguridad dada antes de que pueda ser reutilizada. Si este valor es distinto de cero, entonces SQL Server advertirá al usuario que se está efectuando una copia de seguridad sobre una copia de seguridad ya existente que todavía no ha finalizado su número de días de retención.

Esta configuración resulta útil para los servidores de SQL Server que se encuentren en áreas remotas en las cuales no está disponible un administrador de tiempo completo para gestionar el entorno en lugares en los cuales es probable que el usuario pueda reutilizar cintas de copias de seguridad incorrectamente que debieran de mantenerse intactas durante un cierto periodo.

### **Memory**

Esta opción de configuración controla el máximo número de páginas de 2K de memoria que SQL Server consumirá en el momento del arranque. Para optimizar por completo el servidor para su uso como servidor de bases de datos, es preciso reservar toda la memoria disponible del servidor una vez restados los requisitos mínimos necesarios para Windows NT.

Para ayudar a ajustar la cantidad de memoria que está consumiendo SQL Server, se puede utilizar la orden MEMUSAGE de DBCC, que informará acerca de la forma en que se ha asignado la memoria al servidor y que además mostrará las 20 primeras páginas de datos y de procedimientos almacenados que se hayan ejecutado.

### **Open databases**

Esta opción de configuración controla el número máximo de bases de datos que puede mantener SQL Server en situación abierta en cualquier instante. No debe

---





de establecerse arbitrariamente con un número elevado, porque cada base de datos abierta consume realmente una cierta cantidad de memoria del servidor.

### **Procedure cache**

Controla la proporción de memoria que toma SQL Server con objeto de asignársela a los procedimientos almacenados que se hayan ejecutado más recientemente. Para aquellos sistemas que posean grandes cantidades de procedimientos almacenados, puede ser necesario asignar a este valor algo más de un 30 por ciento, si es que la cantidad total de memoria disponible para SQL Server es relativamente baja.

Se recomienda reducir este valor al 10 por ciento o menos en aquellos sistemas que tengan más de 512 MB de RAM. Es sumamente improbable que la cantidad de procedimientos almacenados en la memoria de reserva superen los 50 MB.

### **Recovery flags**

Controla la información que se visualiza durante el proceso de arranque de SQL Server. Si se le da el valor cero, que es el de default, entonces SQL Server solamente informará que la base de datos se está recuperando o restaurando por su nombre. Si se le da el valor uno, SQL Server informará detalladamente del estado de todas y cada una de las transacciones que estuvieran pendientes en el momento en el que se apagó el servidor y también acerca de la actividad efectuada por SQL Server para resolver este problema.

### **Precovery interval**

Esta opción de configuración controla el número de minutos que requerirá SQL Server para recuperar una base de datos en caso de que se produjera un fallo del sistema de algún tipo. Esta opción, combinada con la cantidad de actividad que se esté produciendo en el servidor, controlará la cantidad de tiempo que transcurra entre CHECKPOINTS de la base de datos.



Un CHECKPOINT puede requerir una cantidad considerable de tiempo si se ha producido una notable actividad en el servidor, pero es frecuente que el uso de puntos de comprobación reduzca la cantidad de tiempo requerida para reiniciar el servidor, porque no tendrá ROLL-FORWARD tanta cantidad de trabajo a partir del registro de transacciones.

Cada una de estas selecciones esta referida al sistema a utilizar, puesto que se trabajará con bases de datos. Y que la gran mayoría de las configuraciones están dirigidas a la protección de la información de las mismas. Punto de gran importancia ya que se manejarán grandes cantidades de información, y estas deben ser resguardadas y respaldadas en su momento, o en un punto en especifico rescatar los procesos cuando algunos de estos se llegan a interrumpir u ocurra algún fallo en el sistema.

### **Tipos de mantenimiento**

El mantenimiento se basa en el cambio que va ligado a la corrección de errores, las adaptaciones que se requieren cuando el entorno del software va evolucionando, y a cambios debidos a las mejoras producidas por los requisitos del cliente, de los cuales se definen los principales tipos de mantenimiento:

**Correctivo.** Es la actividad de corrección de los errores detectados en el uso del sistema. Un fallo ocurre cuando el comportamiento de un sistema es diferente del establecido en la especificación:

- De procesamiento: Salidas incorrectas en un programa.
- Rendimiento: Demasiado tiempo de respuesta.
- Programación: Diseño inconsistente de un sistema.
- Documentación: Diferencias entre la funcionalidad de un programa y el manual de usuario.



**Adaptativo.** Este tipo de mantenimiento consiste en la adaptación del software a cambios en su entorno tecnológico (Hardware o Software), este tipo de mantenimiento responde a:

- Cambio en el sistema operativo.
- Entorno de desarrollo del software (nuevos elementos y herramientas como ODBC).

El mantenimiento adaptativo es cada vez más frecuente debido a la gran velocidad a la que está evolucionando el hardware, nuevos sistemas operativos (Linux) y continuas actualizaciones de los existentes (actualizaciones de seguridad, service packs, etc.)

**Perfectivo.** Los requisitos pueden cambiar con el tiempo para ajustarse a nuevas necesidades o para mejorar las prestaciones actuales. A lo que podemos definir como el conjunto de actividades para mejorar o añadir nuevas funcionalidades requeridas por el usuario. Ajuste a nuevos requisitos.

**Preventivo.** Con este tipo de mantenimiento no se pretende alterar las especificaciones funcionales iniciales del sistema de software. Consiste en modificar el software de forma que se mejoren propiedades tales como la calidad, el mantenimiento, etc., pero sin cambiar las especificaciones iniciales.

**Mantenimiento de sistemas (horizontal).** El área de informática es el área que debe estar facultada para poder definir si se emplea la modalidad de contratación externa para el mantenimiento de los sistemas de información. El mantenimiento de los sistemas es responsabilidad del área de informática, que estará encargada de dar soporte a posibles eventualidades en el desarrollo de las operaciones habituales.



**Mantenimiento de equipo (vertical).** Se definen las siguientes modalidades de mantenimiento preventivo y correctivo del hardware de la empresa. La actualización y sustitución del hardware será evaluado por el área de informática o sistema designado, que se apegarán a los procedimientos establecidos y a las características técnicas definidas por la operabilidad del sistema.

Para el sistema que se está realizando, se utilizará el tipo de mantenimiento correctivo en caso de encontrar algún error y dependiendo de las necesidades del cliente, en caso de requerir alguna mejora del software en su rendimiento, flexibilidad, o la implementación de nuevos requisitos, se podría realizar mantenimiento perfectivo y adaptativo.

### 4.7 LIBERACIÓN DEL SISTEMA

Después de realizar las pruebas al sistema desarrollado, verificar que cumple con los requisitos solicitados y una vez que el usuario acepta que se haga la liberación del mismo, se realiza el contrato de servicio para delimitar las acciones que se llevarán a cabo por parte de Ingeniería de Sistemas a partir de la liberación.

#### **Contrato de servicio**

Acuerdo de prestación de servicios que celebran por una parte el Área de Ingeniería de Sistemas que en lo sucesivo se denominará el “Proveedor” y por otra parte el Coordinador de Campañas del Call Center a quien en lo sucesivo se le denominará el “Cliente”. Ambas partes celebran el presente contrato de servicio en el que se describen responsabilidades específicas y se establece el compromiso al cumplimiento de las mismas a partir de la fecha de firma del presente.



### Responsabilidades generales

#### Del cliente:

- Utilizar los servicios dentro de los horarios convenidos, siendo responsable del manejo adecuado de los mismos y de los equipos a su cargo.
- Reportar cualquier falla referente a la entrega de Servicios y/o Productos a través de los diferentes medios descritos en el anexo A.1. Para los reportes que hayan sido cerrados sin ser atendidos o fallas que se vuelvan a presentar solicitar reapertura de ticket con un máximo de 48 Hrs.
- Solicitar los nuevos requerimientos de desarrollo a través del Ejecutivo de Ingeniería de Sistemas, el cual los analizará de manera conjunta con el Coordinador de Campañas del call center.



#### Del proveedor:

- Garantizar la continuidad operativa de los servicios descritos en el presente contrato.
- Asegurar la funcionalidad de los servicios y/o productos informáticos que se desarrollan y atienden por el Proveedor, mismos que serán definidos en anexos específicos, convenidos en el presente contrato.
- Establecer criterios de seguridad y confidencialidad, para el resguardo de la información.
- Establecer esquemas compatibles entre los procedimientos de reportes y/o registros de fallas para analizar, determinar y corregir las causas raíz de las problemáticas presentadas en el servicio.



- Establecer procesos de afinación del comportamiento y planeación de capacidades del cual participarán como asesores las áreas de desarrollo.
- Mantener informado al Cliente sobre el estado que guardan los principales problemas derivados de la operación diaria, y que afectan de manera significativa los niveles de servicio acordados.
- Enviar en forma mensual los costos a cubrir por el Cliente de acuerdo a volúmenes y productos emitidos, vía transferencia de costos Informáticos, proporcionado(s) por el Cliente y de acuerdo a los costos vigentes emitidos por el Proveedor.

El presente contrato de servicio, surtirá efecto a partir de la fecha en que lo firmen los responsables de cada área, quienes se comprometen a respetar lo establecido en el presente documento, considerándose formalizado con una vigencia permanente hasta nuevo aviso por parte del Cliente o del Proveedor.

### **Anexo A.1 Obligaciones y responsabilidades específicas**

#### **Servicio: atención a usuarios**

Atender los reportes que el Cliente haga a través de los diferentes medios (llamada telefónica, fax, correo electrónico, etc.), ante irregularidades en la correcta entrega de Servicios / Productos ofrecidos y acordados en el presente acuerdo referentes a:

- Fallas de la aplicación o de los servicios (Software)
- Fallas de equipo (PC's, servidores)
- Fallas de comunicación



Proporcionar un número de reporte en un tiempo no mayor a 5 minutos. Informar al Cliente de cualquier comportamiento del servicio que pueda afectar los Niveles de Servicio en este Convenio.

### **Servicio: administración de la operación y respaldos de información**

#### **Es responsabilidad del administrador de los servidores:**

- Atender las fallas de los servidores que le sean asignadas, indicando las causas que generaron el evento, así como la solución que haya sido dada y su tipo, así como escalar los que cuya corrección no correspondan a su función, una vez que hayan sido atendidos de acuerdo a parámetros establecidos y a la tipificación de fallas.
- Será responsable de la generación, custodia y conservación de los respaldos de acuerdo a las siguientes políticas de conservación:

✓ RESPALDOS SEMANALES	5 GENERACIONES
✓ RESPALDOS QUINCENALES	5 GENERACIONES
✓ RESPALDOS MENSUALES	13 GENERACIONES
✓ RESPALDOS ANUALES	12 GENERACIONES
- En caso de existir alguna eventualidad o contingencia, tener los respaldos necesarios para la recuperación de las bases de datos y reportes necesarios para la continuación de los servicios que proporciona la aplicación.
- Mantener la dotación necesaria de cd's y dvd's para la generación de respaldos semanales, quincenales, mensuales, anuales y eventuales de acuerdo a lo especificado en los procesos.



- Deberá garantizar el buen funcionamiento de los servidores, así como la infraestructura de hardware y software bajo su responsabilidad, necesarios para la prestación del servicio.

NOTA: Los CD's o DVD's que contengan los respaldos de la información, deberán almacenarse fuera del inmueble donde se encuentran los servidores para asegurar su correcto resguardo y continuidad del servicio en caso de alguna contingencia. Para lo cual se recomienda el uso de alguna bóveda o caja de seguridad dentro de un banco si no se cuenta con un sitio alternativo para su resguardo.

### **Servicio: desarrollo**

#### **Es responsabilidad de ingeniería de sistemas:**

- Desarrollar toda la presentación, funcionalidad, conectividad y operatividad de los programas del sistema.
- Proporcionar todos los componentes aplicativos y la documentación al momento de efectuar la liberación.
- Atender las solicitudes de nuevos productos y/o modificaciones a los programas, según la naturaleza de la plataforma donde se procesa la información del sistema motivo del Convenio.
- Actualizar los manuales de operación al momento de tener nuevas versiones de programas que impliquen cambios operativos en la aplicación y enviar copias de los manuales de operación a las áreas que administrarán el sistema.
- Atender las fallas del sistema que le sean asignadas a Ingeniería de Sistemas, indicando las causas que generaron el evento, así como la solución que haya sido dada, su tipo, y escalar los que cuya corrección no





correspondan a su función, una vez que hayan sido atendidos de acuerdo a parámetros establecidos y a la tipificación de fallas.

- Una vez firmado el Convenio, verificar diariamente el estado del Sistema/Servicios que se le proporcionan al Cliente, a fin de coordinarse con las áreas involucradas para la solución de problemas, y determinar en caso de atraso, los tiempos de atención y entrega que se requerirán para dichos Servicios/Productos, así como dar seguimiento a los reportes levantados hasta su solución final.
- Recibir del Cliente, como única entidad facultada para tal efecto, los medios magnéticos, electrónicos, fax, instrucciones o datos en papel que por la naturaleza de la aplicación se requieran para su proceso.
- Monitorear que la entrega de servicio se encuentre dentro de los umbrales convenidos promoviendo y dando seguimiento a la recuperación en caso de interrupción del servicio.
- Realizar mediciones de la entrega del servicio, elaborar indicadores del nivel del servicio en forma integral y de cada una de las áreas por las que atraviesa el servicio de acuerdo al flujo de proceso, con el objetivo de detectar comportamientos y niveles de disponibilidad, con el fin de establecer estrategias de mejora y en caso de que existan desviaciones en alguna de las áreas de operación corroborarlo con los responsables para estandarizar criterios.

### FIRMAS

---

**CLIENTE**

---

**PROVEEDOR**

---

**ADMINSTRADOR DE  
SERVIDORES**

# **CONCLUSIONES**



## CONCLUSIONES

Para concluir este trabajo podemos determinar lo siguiente:

- Cada día se vuelve estrictamente necesario el valerse de herramientas basadas en programación para páginas web y bases de datos que nos permitan conocer datos exactos y que almacenen toda la información de una manera confiable y segura concentrándola en un solo lugar.
- Estos sistemas también ofrecen un ahorro en costos y trabajo extra, ya que al automatizar ciertas tareas se evita tener procedimientos humanos que resultan lentos y que a la larga implican gastos adicionales.
- Para el desarrollo del sistema se utilizó la metodología Yourdon / De Marco, que nos ayudó a desarrollar un modelo lógico utilizando diagramas de contexto, DFD's, diccionario de datos, diagramas de procesos y diagrama entidad – relación.
- El sistema cumplió con la labor de poder informar al cliente que es lo que está sucediendo con sus productos en el mercado, todos los datos que se obtienen son reales y precisos, en los cuales podrán basarse para hacer promociones de otras líneas de productos, tanto nuevas como existentes.
- Los datos obtenidos se almacenan en un servidor de base de datos seguro, en donde podrán ser consultados siempre que sea necesario para campañas posteriores.
- Los tipos de prueba que se aplicaron al sistema fueron de caja blanca y de caja negra, en los cuales se probó la funcionalidad del front y del back end para asegurar la consistencia de la información que se captura y se almacena.



- La información generada en el sistema es completamente confiable, ya que se minimizan los errores humanos y todo es consultado por medio de tablas que contienen información específica y bien delimitada.
- El sistema desarrollado tiene una interfaz amigable, que agiliza la captura de información por parte de los operadores de servicio al cliente.
- La aplicación de la metodología de bases de datos y de la ingeniería de sistemas es vital para el desarrollo de cualquier sistema, basándonos en esto podemos estar seguros que nuestro sistema cumple con todas las características para dar respuesta a las necesidades del cliente.
- El sistema funcionará a través de una intranet, desde la cual los usuarios podrán acceder a la información relativa a cada uno de ellos y mostrar las campañas a las cuales estarán asignados, debido a que las pantallas están parametrizadas con los datos más comunes de captura y búsqueda de información.
- Para futuras campañas, se tendrá un repositorio de funciones, clases y objetos que puedan ser reutilizados con los componentes más comunes.
- Para futuras versiones, se tiene planeado migrar de plataforma la capa de presentación del sistema de ASP 3.0 a .NET.
- Con la implantación del sistema, se agilizó la generación y envío de reportes mejorando el desempeño del call center. Se redujeron los tiempos de operación, se mejoraron los niveles de servicio por parte de los operadores y los costos de facturación en el uso de E1's e IVR's disminuyeron.

# **MANUAL TÉCNICO**



## MANUAL TÉCNICO

### Diagramas del sistema

En las Figuras 1 y 2 se muestran los diagramas de arquitectura y de conectividad del Sistema de atención a clientes para el call center.

Para acceder al sistema se necesita una PC que tenga instalado Internet Explorer. La aplicación se instalará en un servidor web (IIS V5.0), la base de datos se instalará en otro servidor (SQL Server 2000), y finalmente el respaldo de la información (BD y reportes) se realizará en CD's o DVD's y en un servidor dedicado especialmente para esta tarea.

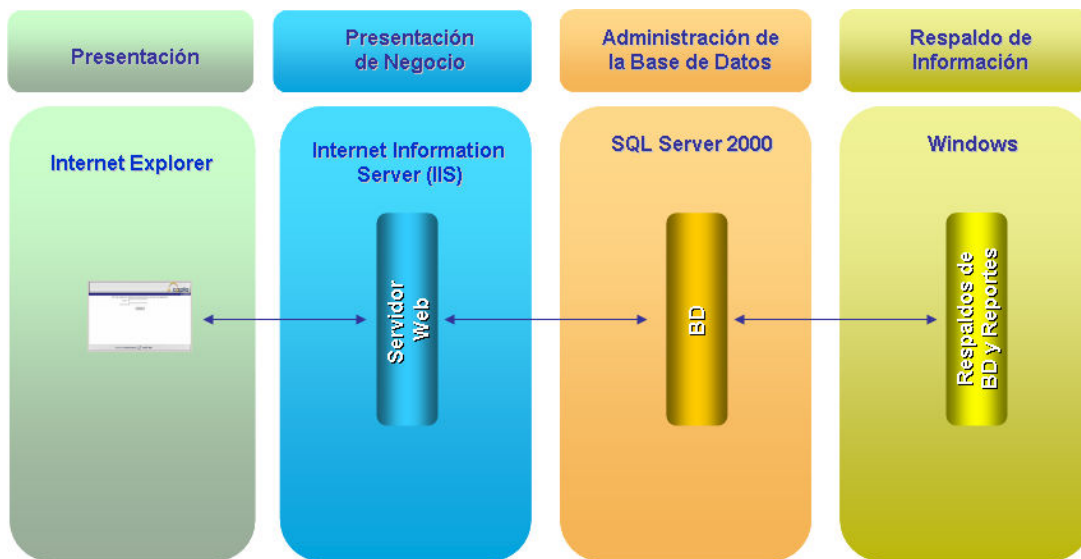


Figura 1. Diagrama de arquitectura del sistema de atención a clientes

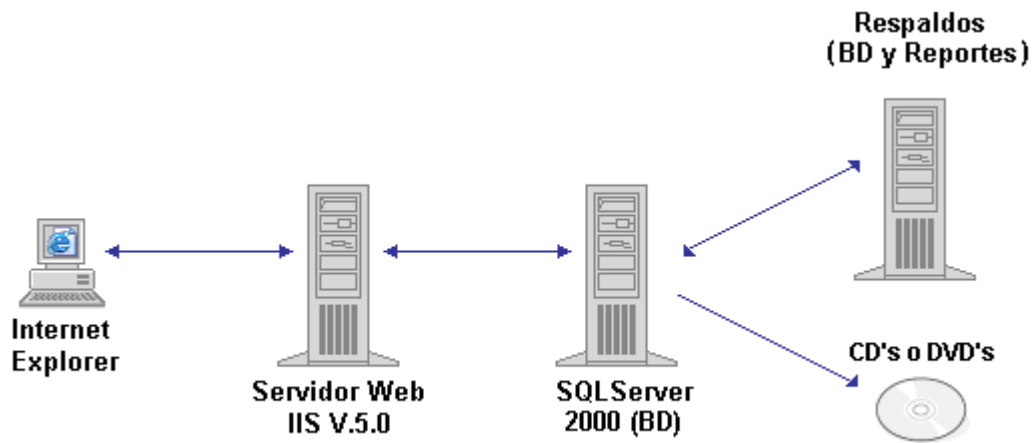


Figura 2. Diagrama de conectividad del sistema de atención a clientes

### Requisitos mínimos para la instalación del sistema de atención a clientes

- Sistema Operativo Windows 2000 ó Windows XP
- Internet Information Server (IIS) 5.1 (servidor web)
- Microsoft SQL Server 2000
- 128MB de RAM

El sistema de atención a clientes está compuesto por tres capas:

- Presentación. Interfaz gráfica que permite acceder al sistema a través de un browser (Internet Explorer).
- Presentación de Negocio. Aplicación ASP en la que está implementada la lógica de negocio.
- Datos. Base de datos del sistema distribuida en una copia de seguridad de SQL Server 2000.



En la Figura 3 se presenta el diagrama general del sistema, donde se muestra el servidor web que contiene las páginas del sistema (Web Server IIS) y el servidor de base de datos (SQL Server 2000) en donde se manejan las reglas de negocio y los datos.

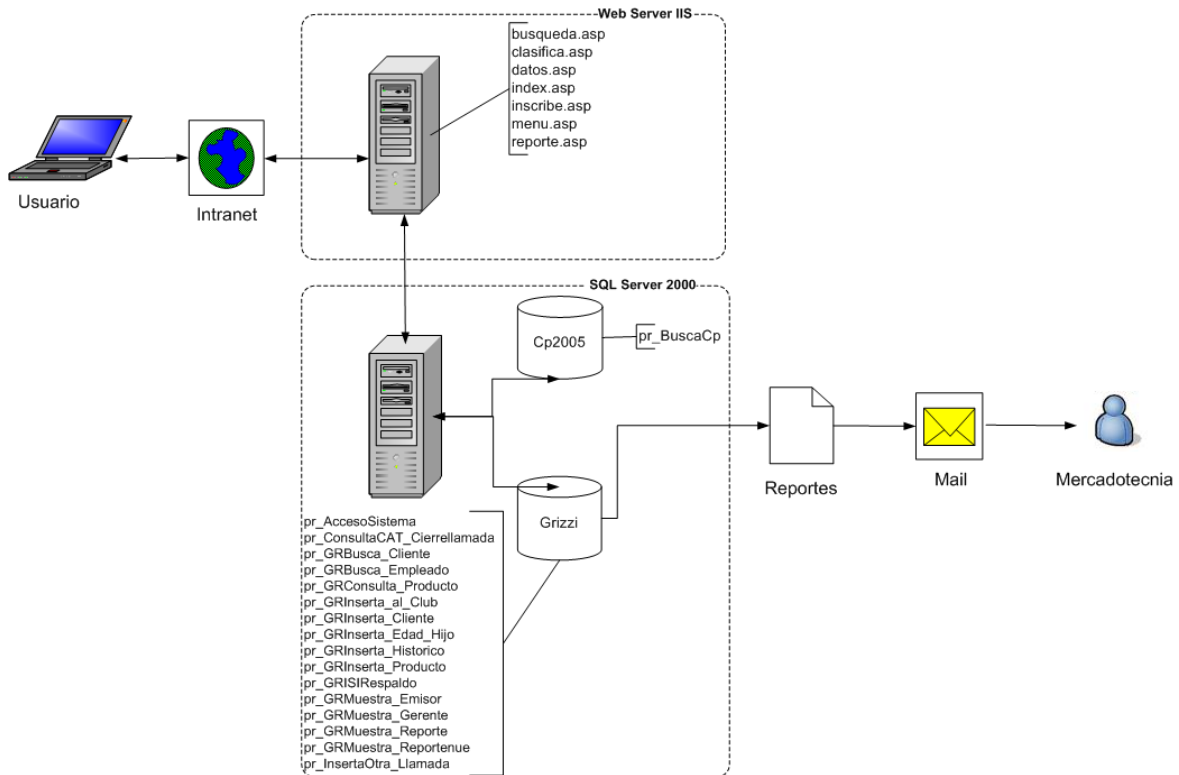


Figura 3. Diagrama general del sistema de atención a clientes

### Instalación de IIS en Windows XP Professional

Internet Information Server (IIS) es el servidor de páginas web de la plataforma Windows. Se distribuye gratuitamente junto con las versiones de Windows basadas en NT, como pueden ser Windows 2000 Professional o Windows 2000 Server, así como Windows XP, también en sus versiones Professional y Server. Estas normas de instalación son aplicables a nivel general para poder cargarlo en nuestro sistema:





1. Inserte el disco de Windows y en la ventana de autoarranque que se muestra seleccione la opción de **Instalar componentes opcionales de Windows**. (Ver Figura 4).



**Figura 4. Pantalla de autorranque de Windows XP**

2. Abra el panel de control y seleccione la opción **Agregar o quitar programas**, después, seleccione el icono de la izquierda marcado como **Agregar o quitar componentes de Windows**. (Ver Figura 5)

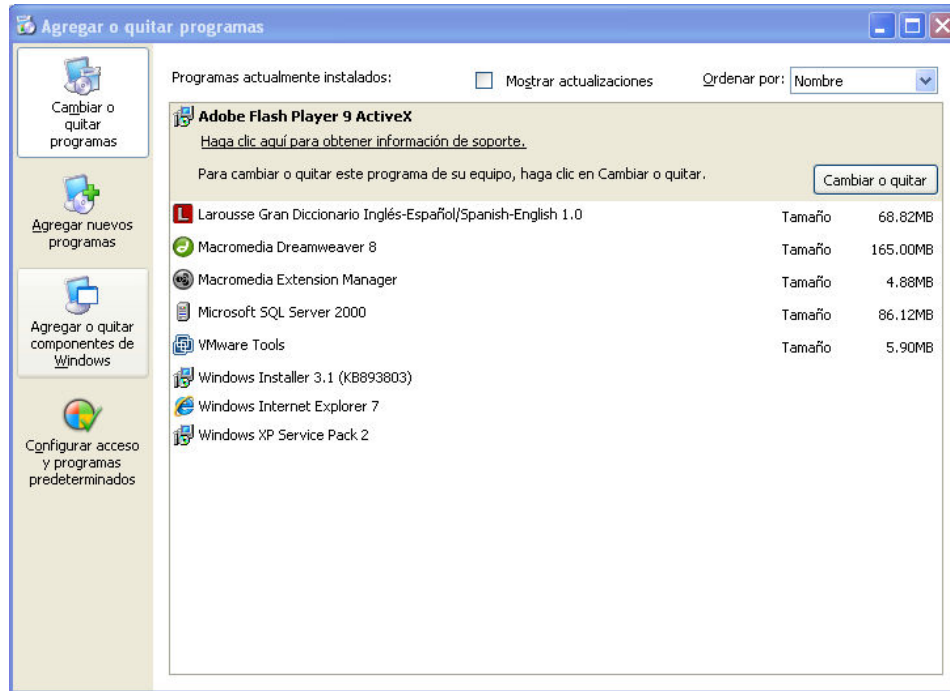
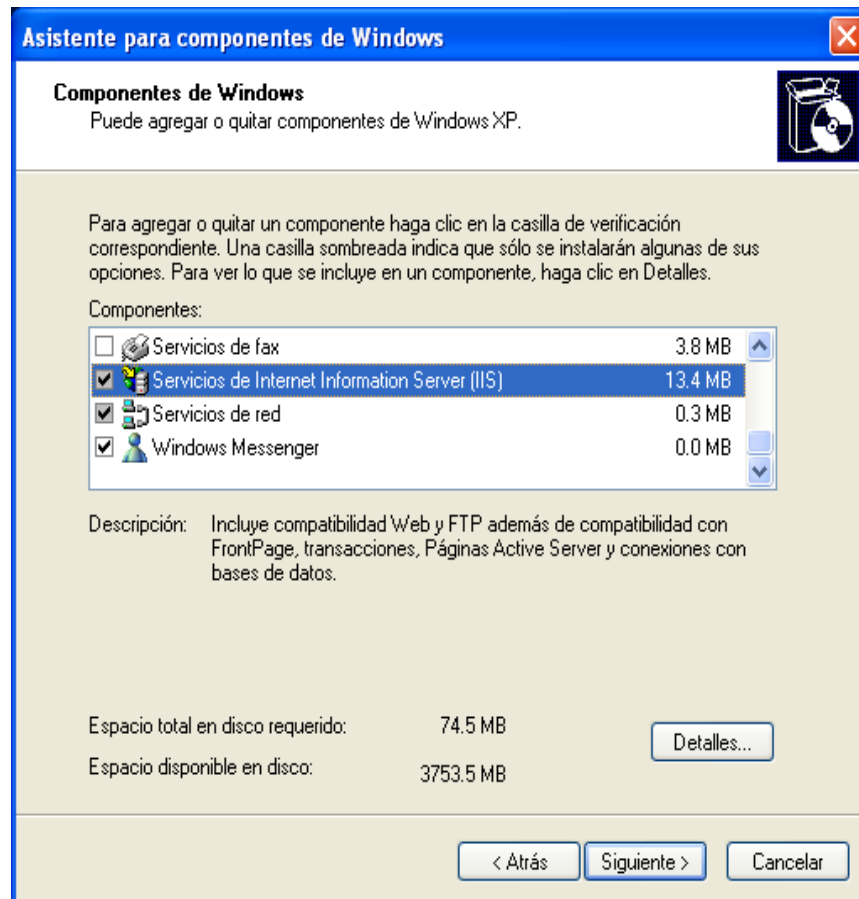


Figura 5. Pantalla para agregar o quitar programas

Se mostrará una ventana en la cual se seleccionan los componentes adicionales de Windows que están disponibles. Marque la opción **Servicios de Internet Information Server** y oprima el botón **Siguiente** para comenzar la instalación. (Ver Figura 6).



**Figura 6. Pantalla para agregar o quitar componentes de Windows XP**

3. Para comprobar si se instaló correctamente IIS, acceda al servidor web, abra una pantalla de Internet Explorer y escriba la dirección <http://localhost> , deberá aparecer una página web informando que IIS está correctamente instalado. (Ver Figura 7).

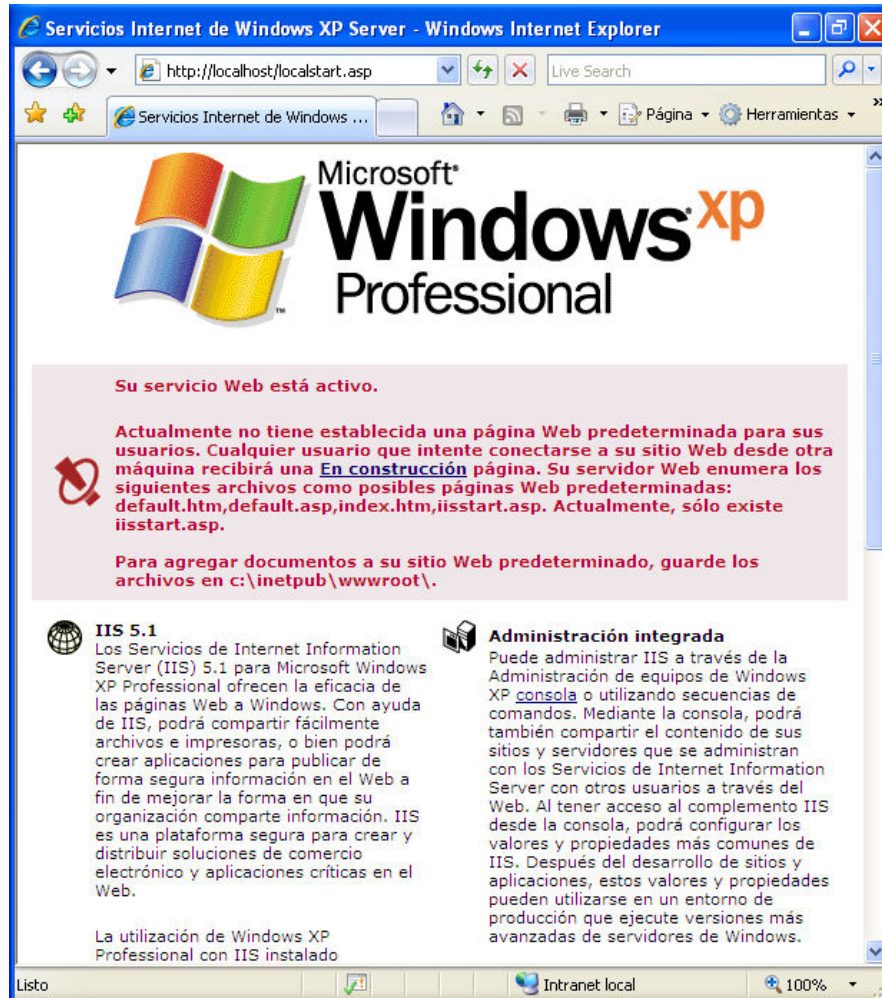


Figura 7. Pantalla que indica que IIS fue instalado correctamente

Lo que se muestra cuando accedemos a <http://localhost> , es el sitio web predeterminado que se guarda en nuestro disco duro en la carpeta C:\inetpub\wwwroot. Si accedemos a dicha carpeta, podremos ver los archivos que están sirviendo como sitio web predeterminado. Encontraremos, entre otros archivos uno llamado “iisstart.asp” que es el que se ejecuta al acceder a este directorio.



## Restauración de la base de datos

Para restaurar la base de datos del sistema siga los siguientes pasos:

1. Entre al programa SQL Server Enterprise Manager (Administrador Corporativo)
2. En la **console root (raíz de la consola)** seleccione la opción **Microsoft SQL Servers (Servidores Microsoft SQL Server) -> SQL Server Group (grupo de SQL Server) -> (local)(Windows NT) -> Databases (bases de datos)**.
3. De click con el botón derecho del mouse y seleccione la opción **Todas las tareas -> Restore database (restaurar base de datos)**.

Se desplegará una ventana en la que se tienen que establecer los parámetros necesarios para restaurar la base de datos. (Ver Figura 8).

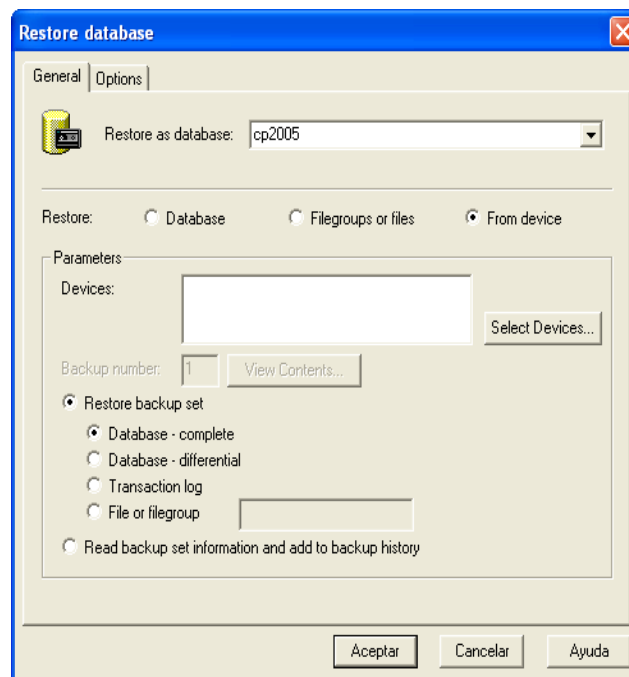


Figura 8. Pantalla para restaurar la base de datos

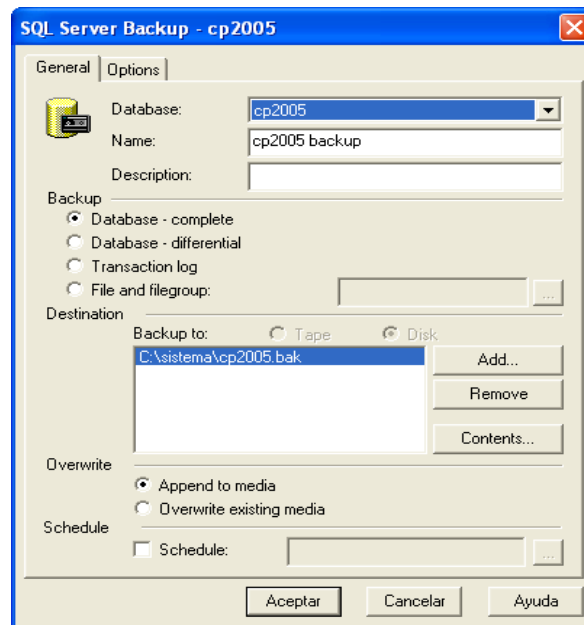


4. En la opción **Restore as database (Restaurar como base de datos)** escriba el nombre de la base de datos del sistema, para nuestro sistema escriba Grizzi.
5. En la opción **Restore (Restaurar)** seleccione la opción **From device (Desde dispositivo)** y presione el botón **Select devices...(Seleccione dispositivos...)**, seleccione la ruta donde se encuentra el archivo que contiene la copia de seguridad de la base de datos del sistema.
6. Presione el botón **Aceptar** y automáticamente se restaurará la base de datos.

### **Copia de seguridad de la base de datos**

Las copias de seguridad de la base de datos se deben realizar constantemente para asegurar la información en caso de alguna contingencia. Para lo cual se deben seguir los pasos siguientes:

1. Entre al programa SQL Server Enterprise Manager (Administrador Corporativo).
2. En la **console root** (raíz de la consola) seleccione la opción **Microsoft SQL Servers** (Servidores Microsoft SQL Server) -> **SQL Server Group** (Grupo de SQL Server) -> (local)(Windows NT) -> **Databases** (Bases de Datos).
3. De click con el botón derecho del mouse y seleccione la opción **Todas las tareas -> Backup database (Respaldo/Copia de seguridad de la base de datos)**. Se desplegará la pantalla en la que se deben proporcionar los parámetros necesarios para la creación del respaldo de la base de datos. (Ver Figura 9).



**Figura 9. Pantalla del SQL Server Enterprise Manager**

4. Seleccione el nombre de la base de datos en el combo database a la cual se le hará una copia de seguridad (backup).
5. En el campo **Name** escriba el nombre que se le dará a la copia de seguridad
6. En el campo **Description** escriba una descripción de la copia de seguridad
7. Backup (copia de seguridad) indica el tipo de copia de seguridad que se desea realizar, seleccione la opción **Database complete** (base de datos completa).

Existen varios tipos de copias de seguridad:

- **Database - complete (base de datos completa).** Esta copia incluye todos los objetos de la base de datos (tablas, vistas, store procedures, triggers).

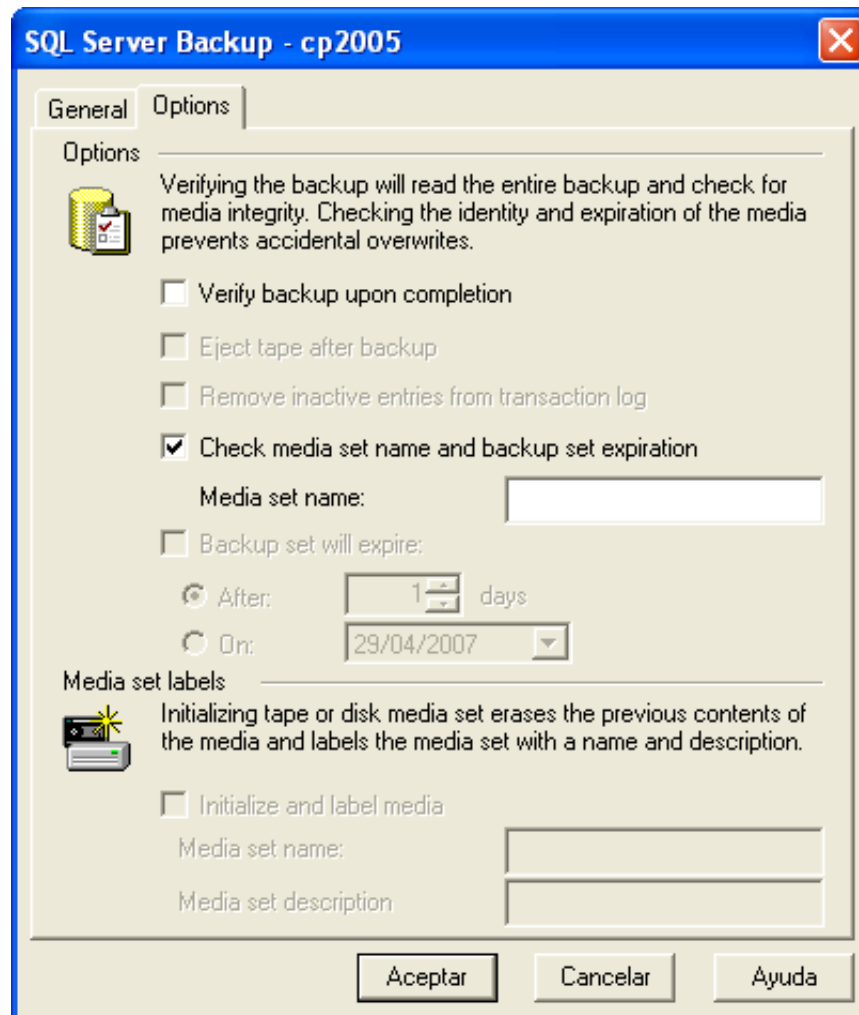


- **Database - differential (base de datos diferencial).** Solo copia las modificaciones que se le hayan hecho a la base de datos (DML, DDL) desde la última copia de seguridad completa.
  - **Transaction log (registro de transacciones).** Copia del registro de transacciones, incluye todas las transacciones realizadas por los usuarios (inserts, updates, deletes), al tener toda esta información, se puede recuperar todo lo que hayan hecho los usuarios.
  - **File and filegroup (archivo y grupo de archivos).** Permite realizar una copia de seguridad de una porción de la base de datos. Este tipo de copia de seguridad se debe utilizar cuando la base de datos es muy grande y también se debe realizar la copia del registro de transacciones para poder recuperarse.
8. **Destination (destino).** En este campo se indica la ruta y nombre del archivo donde se desea almacenar la copia de seguridad, utiliza el botón Add.. para seleccionar la ruta deseada.
9. **Overwrite (sobrescribir).** Este campo permite seleccionar si la información de la copia de seguridad se agrega a una copia existente o si se sobrescribe. Seleccione la opción **Overwrite existing media (sobrescribir medio existente).**
10. Cámbiese a la pestaña **Options (opciones)** y seleccione la opción **Verify backup upon completion (Comprobar la copia de seguridad al concluir)** y regrese a la pestaña **General.** (Ver Figura 10).





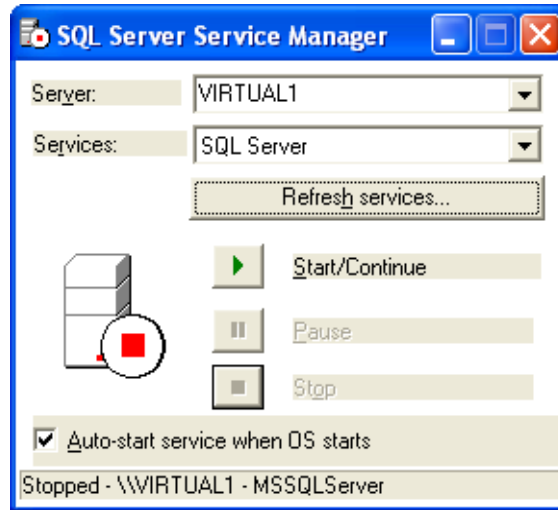
11. Presione el botón **Aceptar** para crear la copia de seguridad del sistema con los parámetros seleccionados.



**Figura 10. Pantalla para realizar copia de seguridad de la base de datos**

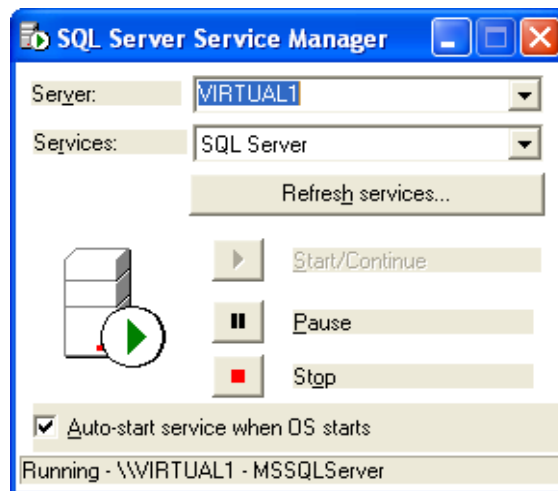
### **Levantar la base de datos en caso de que se caiga o esté detenida**

En caso de que la base de datos esté apagada o se caiga, tiene que entrar al programa SQL Server Service Manager y verificar en que status se encuentra, si al abrir el programa se ve un cuadro rojo junto al icono del servidor, presione el botón **Start / Continue** para levantarlo. (Ver Figura 11).



**Figura 11. Pantalla para levantar la base de datos**

Al terminar de levantarse la base de datos, se debe ver un triángulo verde junto al icono del servidor, esto indicará que la base de datos se levantó sin problemas. (Ver Figura 12).



**Figura 12. Pantalla para tirar la base de datos**

Si la base de datos no se levantó correctamente, vuelva a oprimir el botón de **Start / Continue (Levantar/Continuar)**.



### Listado de páginas y stored procedures

En la tabla 1 se muestra el listado de páginas y programas que se encuentran en el servidor, la descripción de cada uno de ellos y la ruta donde están instalados.

Archivo	Descripción	Ruta
index.asp	Inicio de sesión	\\sistema\grizzi
busqueda.asp	Busqueda de clientes	\\sistema\grizzi
datos.asp	Captura y modificación de registro	\\sistema\grizzi
clasifica.asp	Clasificación de productos	\\sistema\grizzi
inscribe.asp	Inscripción al club	\\sistema\grizzi
menu.asp	Acceso de administración	\\sistema\grizzi
reporte.asp	Envío de reportes	\\sistema\grizzi
date-picker.js	Script de calendario	\\sistema\grizzi\tools
db.asp	Presentación de header y footer	\\sistema\grizzi\tools
funciones.asp	Funciones generales del sistema	\\sistema\grizzi\tools
styles.css	Hoja de estilos	\\sistema\grizzi\tools
valida.js	Validaciones en java script	\\sistema\grizzi\tools

**Tabla 1. Listado de programas del sistema de atención a clientes**



En la tabla 2 se muestran los stored procedures del sistema, las tablas con las que se relacionan y las páginas ASP donde se utilizan.

Nombre del stored procedure	Tabla relacionada	Página
pr_AccesoSistema	GRIZZI_EMPLEADOS	index.asp
pr_InsertaOtra_Llamada	CIERRA_LLAMADA OTRA_LLAMADA	busqueda.asp datos.asp clasifica.asp inscribe.asp
pr_GRBusca_Cliente	CAMP_GRIZZI	busqueda.asp datos.asp
pr_ConsultaCAT_Cierrellamada	CIERRA_LLAMADA	busqueda.asp datos.asp clasifica.asp inscribe.asp
pr_GRInserta_Cliente	CAMP_GRIZZI	datos.asp
pr_GRInserta_Producto	CAMP_GRIZZI CAT_PRODUCTO_GRIZZI CIERRA_LLAMADA GRIZZI_CLIENTE_PRODUCTO	clasifica.asp
pr_GRBusca_Empleado	GRIZZI_EMPLEADOS	menu.asp
pr_GRInserta_al_Club	CAMP_GRIZZI GRIZZI_CLUB_RICITOS	inscribe.asp
pr_GRInserta_Edad_Hijo	GRIZZI_HIJO	inscribe.asp

**Tabla 2. Listado de stored procedures del sistema de atención a clientes**

En la Tabla 3 se despliegan los posibles mensajes o errores que pueden enviar los stored procedures.



Stored procedure	Mensajes / Errores
pr_AccesoSistema	USUARIO O PASSWORD INVALIDO
pr_InsertaOtra_Llamada	ESE CIERRE DE LLAMADA NO ES VALIDO
	EL OPERADOR NO ES VALIDO
	<<ERROR>> AL INSERTAR EN OTRA_LLAMADA
	SE HA INSERTADO OTRA LLAMADA EXITOSAMENTE
pr_GRBusca_Cliente	NO HAY SUFICIENTES PARAMETROS PARA HACER UNA BUSQUEDA
	<<ERROR>>, NO EXISTE ESE CLIENTE
	<<ERROR>>, NO EXISTEN CLIENTES CON ESE NOMBRE (1)
	<<ERROR>>, NO EXISTEN CLIENTES CON ESE APELLIDO PATERNO
	<<ERROR>>, NO EXISTEN CLIENTES CON ESE APELLIDO MATERNO
	<<ERROR>>, NO EXISTEN CLIENTES CON ESE APELLIDO
	<<ERROR>>, NO EXISTEN CLIENTES CON ESE TELEFONO
pr_ConsultaCAT_Cierrellamada	NO EXISTEN DATOS A MOSTRAR. VERIFIQUE
pr_GRInserta_Cliente	EL CLIENTE YA FUE REGISTRADO, REALICE UNA BUSQUEDA
	EL TELEFONO NO ES VALIDO, VERIFIQUE
	EL NUMERO DE HIJOS NO ES VALIDO
	<<ERROR>> AL INSERTAR EN CAMP_GRIZZI
pr_GRInserta_Producto	EL CLIENTE NO EXISTE
	EL PRODUCTO NO EXISTE
	EL CIERRE DE LLAMADA NO ES VALIDO
	<<ERROR>> AL INSERTAR EN GRISI_PRODUCTO
pr_GRBusca_Empleado	NO HAY SUFICIENTES PARAMETROS PARA HACER UNA BUSQUEDA
	<<ERROR>>, NO EXISTE EL EMPLEADO
	<<ERROR>>, NO EXISTEN EMPLEADOS CON ESE NOMBRE (1)
	<<ERROR>>, NO EXISTEN EMPLEADOS CON ESE APELLIDO PATERNO
	<<ERROR>>, NO EXISTEN EMPLEADOS CON ESE APELLIDO MATERNO
	<<ERROR>>, NO EXISTEN EMPLEADOS CON ESE NOMBRE
	<<ERROR>>, NO EXISTEN EMPLEADOS CON ESE STATUS
	<<ERROR>>, NO EXISTEN EMPLEADOS DE ESE TIPO
	<<ERROR>>, NO EXISTEN EMPLEADOS CON ESE APELLIDO



pr_GRInserta_al_Club	EL CLIENTE NO EXISTE
	EL CLIENTE YA ESTA EN EL CLUB
	EL NOMBRE DEL BEBE NO PUEDE ESTAR VACIO
	EL OPERADOR YA NO TRABAJA AQUI
	<<ERROR>> AL INSERTAR AL CLUB RICITOS
	<<ERROR>> AL ACTUALIZAR EL CLIENTE
pr_GRInserta_Edad_Hijo	EL CLIENTE NO EXISTE O SE INDICO QUE NO TIENE HIJOS
	LA EDAD NO ES VALIDA
	YA SE INSERTO EL NUMERO DE HIJOS INDICADO, VERIFIQUE
	<<ERROR>> AL INSERTAR EL HIJO EN GRIZZI_HIJO
	EL HIJO SE INSERTO CORRECTAMENTE

**Tabla 3. Listado de mensajes / errores por stored procedure**

En la Figura 13 se presenta el diagrama completo del sistema mostrando la relación de todas las paginas ASP, con los stored procedures y las tablas que utilizan.

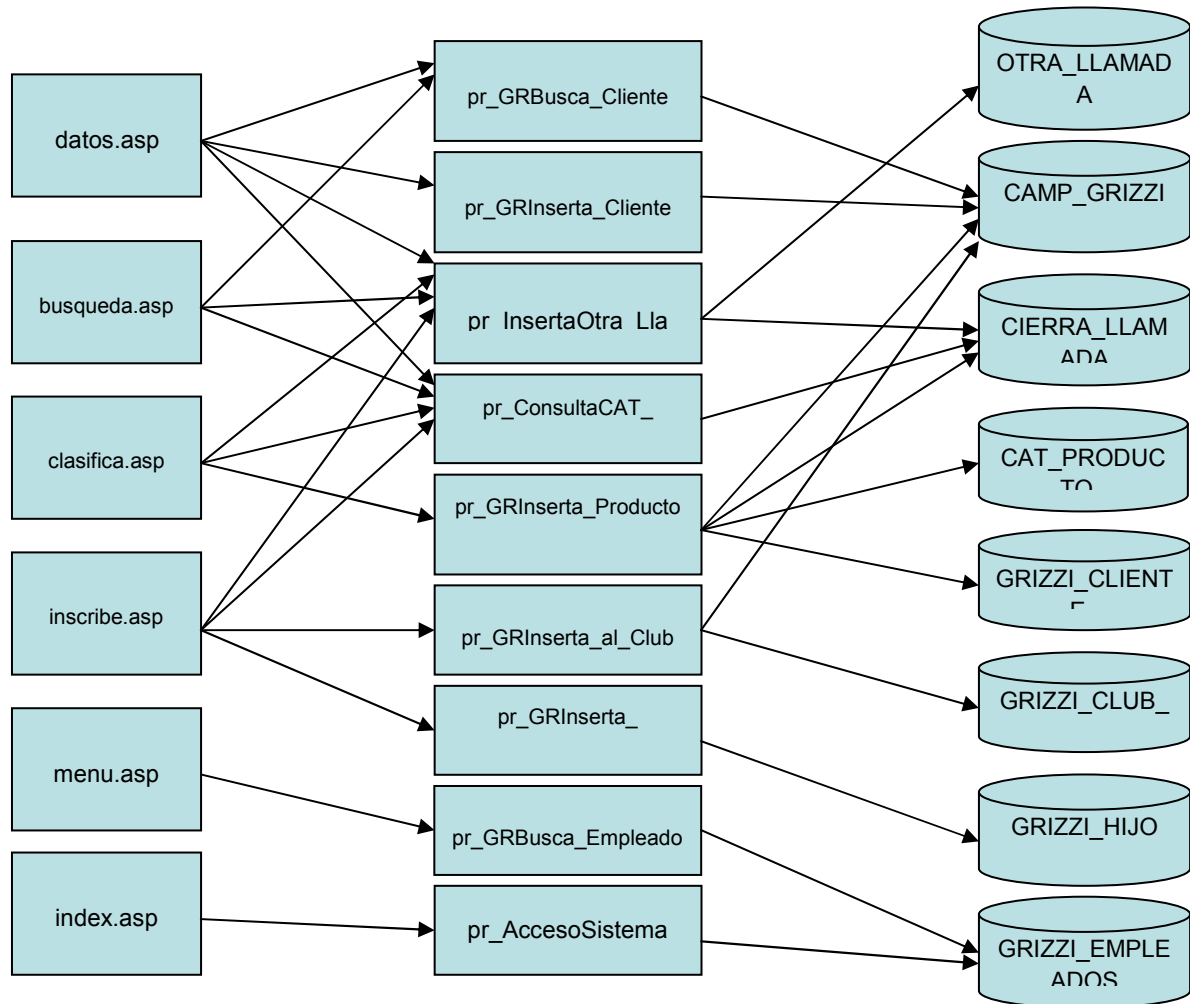


Figura 13. Diagrama general del sistema de atención a clientes

# **MANUAL DE USUARIO**





## MANUAL DE USUARIO

El objetivo que se persigue con la aplicación del presente manual es dar a conocer a los usuarios finales las características y las formas de funcionamiento del software Intranet.

### Autenticación

El ingreso a la aplicación es a través de la Intranet cuya dirección electrónica es: <http://intranet/campanias/grisi/>

En la Figura 1 se muestra la pantalla de ingreso al sistema, escriba el "Usuario" y "Password" y presione el botón **Entrar**.

Por favor ingrese la siguiente información para acceder a la aplicación.

Usuario:

Password:

Figura 1 Pantalla de logueo

### Búsqueda

Una vez que se halla logueado dentro del sistema se desplegará la pantalla de la Figura 2, que corresponde al módulo de búsqueda de clientes:

Búsqueda del Cliente

No. Contacto :

Nombre :  A. Paterno :

Teléfono :

Figura 2 Búsqueda de clientes



Aquí se realizan las búsquedas dentro del sistema, ya sea por nombre de contacto, teléfono y por nombre con apellido.

Si la búsqueda que ha realizado es exitosa, según el parámetro que haya ingresado, se mostrará la Figura 3.

**Búsqueda del Cliente**

No. Contacto :

Nombre :  A. Paterno :

Teléfono :

**Resultados de la búsqueda**

No. Cte.	Nombre	Teléfono	Estado	Inscrito
1	Alejandro Fiesco	(223) 25123685	Puebla	No
2	Araceli Velazquez	(223) 25128585	Puebla	No
3	Alejandro Fiesco	(223) 25123685	Puebla	No
4	Araceli Velazquez	(223) 25128585	Puebla	No
5	Alejandro Fiesco	(223) 25123685	Puebla	No

**Figura 3 Resultados de búsqueda**

Como podrá observar debajo de los campos de búsqueda, aparece un botón llamado “Nuevo registro”. Este botón es útil para cuando desea dar de alta un nuevo usuario que no apareció en la lista de los resultados de la búsqueda. Basta con hacer click en dicho botón para realizar la captura de los datos del cliente.

Si la búsqueda es exitosa haga click en cualquier campo correspondiente al renglón de la lista y le llevará a la captura de los datos del cliente. (Ver Figura 4).





## Inscripción al club ricitos

Si durante la llamada el cliente mencionó su inscripción al club aparecerá la pantalla que se muestra en la Figura 5.

<b>Datos Conyuge</b>	
Nombre: <input type="text" value="ABEL"/>	Ap. Paterno: <input type="text" value="SAGREO"/>
Ap. Materno: <input type="text" value="REYES"/>	
<b>Datos Bebé</b>	
<input type="radio"/> Nacido <input type="radio"/> Por Nacer	
Nombre: <input type="text" value="ABEL"/>	Ap. Paterno: <input type="text" value="SAGREO"/>
Ap. Materno: <input type="text" value="REYES"/>	
Fecha de Nac.: <input type="text" value="12"/> <input type="text" value="01"/> <input type="text" value="2002"/>	Sexo: <input type="radio"/> Femenino <input type="radio"/> Masculino
<b>Otros Hijos</b>	
	Edad                      ¿Cuántos Hijos?
<input type="checkbox"/>	1 <input type="text"/>
<input type="checkbox"/>	2 <input type="text"/>
<input type="checkbox"/>	3 <input type="text"/>
<input type="checkbox"/>	4 <input type="text"/>
<input type="checkbox"/>	5 <input type="text"/>
<input type="checkbox"/>	6 <input type="text"/>
<input type="checkbox"/>	De 7 a 10 <input type="text"/>
<input type="checkbox"/>	Mas de 10 <input type="text"/>

GUARDAR

**Figura 5 Inscripción al club ricitos**

Llene los campos necesarios y termine haciendo clic en el botón de “Guardar”

## Clasificación de la llamada

La pantalla mostrada en la Figura 6 aparecerá una vez que el cliente se haya o no inscrito al club, llene los campos que se muestran y termine dando un click en el botón “Guardar”.



**Clasificación de la Llamada**

Línea:

Producto:

Subproducto:

Tipificación:

Lugar de Compra:

Comentarios:

GUARDAR

**Figura 6 Clasificación de la llamada**

Si todo el proceso se realizó de forma correcta, se despliega el mensaje mostrado en la Figura 7 y se regresará a la pantalla de búsqueda de clientes.



**Figura 7 Proceso exitoso**

### Casos especiales

Si durante la sesión tuvo problemas en la llamada, puede tipificarlos en la sección de cierre de llamadas mostrada en la Figura 8. Si es necesario agregue algún tipo de comentarios



Cierres de Llamada

Tipo de cierre:

REGISTRAR LLAMADA

**Figura 8 Cierres de llamada**

Para terminar su sesión dentro del sistema haga click en el botón **Cerrar** que se encuentra debajo del logotipo y que le acompaña durante toda su sesión de trabajo. (Ver Figura 9).

CERRAR

Búsqueda del Cliente

No. Contacto :

Nombre :  A. Paterno :

Teléfono :

**Figura 9 Cierre de sesión**

# **GLOSARIO DE TÉRMINOS**



## GLOSARIO DE TÉRMINOS

### A

**Aplicación Web.** Sistema informático que los usuarios utilizan accediendo a un servidor web a través de Internet o de una intranet. Las aplicaciones web son populares debido a la practicidad del navegador web como cliente ligero. La habilidad para actualizar y mantener aplicaciones web sin distribuir e instalar software en miles de potenciales clientes es otra razón de su popularidad.

### B

**Back-end.** Es la parte de un sistema informático que procesa la entrada desde el front-end.

### C

**Cliente-Servidor (Modelo).** Esta arquitectura consiste básicamente en que un programa, el Cliente informático realiza peticiones a otro programa, el servidor, que les da respuesta. Aunque esta idea se puede aplicar a programas que se ejecutan sobre una sola computadora es más ventajosa en un sistema multiusuario distribuido a través de una red de computadoras.

En esta arquitectura la capacidad de proceso está repartida entre los clientes y los servidores, aunque son más importantes las ventajas de tipo organizativo debidas a la centralización de la gestión de la información y la separación de responsabilidades, lo que facilita y clarifica el diseño del sistema.

### D

**DNIS (Dialed Number Identification Service).** Números Telefónicos enviados de la red telefónica pública para identificar el número telefónico que está realizando la llamada.





### E

**E-Mail.** El correo electrónico, en inglés electronic email o e-mail, es un método para crear, enviar y recibir mensajes a través de sistemas de comunicación electrónica.

### F

**Front-end.** Es la parte del software que interactúa con el usuario. Es el responsable de recolectar los datos de entrada del usuario, que pueden ser de muchas y variadas formas y procesarlas de una manera conforme a la especificación que el back-end pueda usar.

### I

**Inbound Calls (Llamadas de Inbound).** Llamadas realizadas en el call center .

### M

**Marketing.** Es el conjunto de principios, metodologías y técnicas a través de las cuales se busca la obtención de objetivos de negocio, mediante el diseño y aplicación de estrategias para satisfacer las necesidades y los requerimientos de los clientes y los consumidores. Es la principal herramienta de la gestión comercial.

### O

**OLAP (*On-Line Analytical Processing*).** Es una solución utilizada en el campo de la Inteligencia de Negocios (Business Intelligence), la cual consiste en consultas a estructuras multidimensionales (o Cubos OLAP) que contienen datos resumidos de grandes Bases de Datos o Sistemas Transaccionales (OLTP). Se usa en informes de negocios de ventas, marketing, informes de dirección, minería de datos y áreas similares.

**Outbund Calls (Llamadas Outbound).** Recepción de llamadas en el call center.



### P

**Private Branch Exchange (PBX).** Es un servicio ofrecido por una empresa de telecomunicaciones, por el cual una cantidad  $n$  de líneas o números son agrupadas en un único número que se publica o muestra al público y al cual pueden llamar. La empresa proveedora se encarga de distribuir las llamadas entrantes por las líneas disponibles contratadas por el cliente (empresa privada, ONG, ORG, Gobierno, escuelas, etc.).

### R

**RDBMS (Relational Data Base Management System).** Es un Sistema Administrador de Bases de Datos Relacionales que proporcionan el ambiente adecuado para gestionar una base de datos.

### S

**Secure Sockets Layer.** Son protocolos criptográficos que proporcionan comunicaciones seguras en Internet. SSL proporciona autenticación y privacidad de la información entre extremos sobre Internet mediante el uso de criptografía. Habitualmente, sólo el servidor es autenticado (es decir, se garantiza su identidad) mientras que el cliente se mantiene sin autenticar; la autenticación mutua requiere un despliegue de infraestructura de claves públicas (o PKI) para los clientes. Los protocolos permiten a las aplicaciones cliente-servidor comunicarse de una forma diseñada para prevenir escuchas, la falsificación de la identidad del remitente y mantener la integridad del mensaje.

**Servidor Web.** Es un programa que implementa el protocolo HTTP (hypertext transfer protocol). Este protocolo está diseñado para transferir lo que llamamos hipertextos, páginas web o páginas HTML (hypertext markup language): textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de música.



### T

**TCP/IP (Transmission Control Protocol/Internet Protocol).** Se trata de un estándar de comunicaciones muy extendido y de uso muy frecuente para software de red basado en Unix con protocolos Token-Ring y Ethernet, entre otros. Es compatible con productos de muchas marcas: IBM, DEC, Sun, AT&T, Data General, etc. TCP/IP es conforme a los niveles 3 y 4 de los modelos OSI. Este conjunto de protocolos fue desarrollado originalmente para el Departamento de Defensa de Estados Unidos.

### W

**WWW (World Wide Web).** Es un medio de comunicación de texto, gráficos y otros objetos multimedia a través de Internet, es decir, la web es un sistema de hipertexto que utiliza Internet como su mecanismo de transporte o desde otro punto de vista, una forma gráfica de explorar Internet.

# **BIBLIOGRAFÍA**



## BIBLIOGRAFÍA

- Jesús Bobadilla y Alejandro Alcocer, ASP Active Server Pages 3.1, Ed. McGraw Hill Osborne.
- Christian Wenz y Christian Trennhaus, ASP Active Server Pages, Marcombo Boixareu editores.
- Paul Whitehead, Active Server Pages 3.0, ST Editorial Inc.
- Derek Ferguson, ASP sin errores, Ed. McGraw Hill Osborne.
- Pressman, Roger S., Ingeniería de Software un enfoque práctico, McGrawHill, 1993.
- [http://www.unalmed.edu.co/~mstabare/disenio\\_conceptual.htm](http://www.unalmed.edu.co/~mstabare/disenio_conceptual.htm), visitado el 17 de marzo de 2007.
- <http://tramullas.com/documatica/2-7.html>, visitado el 17 de marzo de 2007.
- [http://www.wikilearning.com/gastos\\_indirectos\\_de\\_fabricacion\\_costos\\_de\\_calidad\\_y\\_gastos\\_de\\_distribucion\\_y\\_administracion-wkccp-13005-1.htm](http://www.wikilearning.com/gastos_indirectos_de_fabricacion_costos_de_calidad_y_gastos_de_distribucion_y_administracion-wkccp-13005-1.htm), visitado el 20 de marzo de 2007.
- <http://www.imt.com.mx/recontact/11/index.php>, visitado el 31 de marzo de 2007
- <http://reyporundia.phplibre.com.es/personal/apuntes/Apuntes%20de%20FIS.pdf>, visitado el 31 de marzo de 2007.
- <http://guaica.uniandes.edu.co:5050/dspace/handle/1992/823>, 5 abril 2007.
- [http://notes.sistema.itesm.mx/apps/viti/bb/cursos.nsf/018b1c7685031ca686256c610080a1bd/16be6ebdfaf78fdc86256f5f005e48a2/\\$FILE/Presentation%20Seminario%20-Proceso%20BB.ppt](http://notes.sistema.itesm.mx/apps/viti/bb/cursos.nsf/018b1c7685031ca686256c610080a1bd/16be6ebdfaf78fdc86256f5f005e48a2/$FILE/Presentation%20Seminario%20-Proceso%20BB.ppt), visitado el 5 abril 2007.
- <http://mx.geocities.com/alfonsoaraujocardenas/topologias.html>, visitado el 6 abril 2007.



- [http://www.mundo-contact.com/Acervo/m1999\\_evaluacion.pdf](http://www.mundo-contact.com/Acervo/m1999_evaluacion.pdf), 5 abril 2007.
- <http://www.geocities.com/TimesSquare/Chasm/7990/topologi.htm>, visitado el 6 abril 2007.
- <http://www.ingenieroguzman.com.ar/notas/tcpip/encapsulado.html>, visitados el 6 abril 2007.
- <http://www.networksorcery.com/enp/protocol/tcp.htm>, visitado el 6 abril 2007
- <http://www.it.uc3m.es/vlc/tesis.pdf>, 8 abril 2007.
- [http://www.e-contact.com.mx/Acervo/m2003\\_DelaEstrategia.pdf](http://www.e-contact.com.mx/Acervo/m2003_DelaEstrategia.pdf), visitado el 14 abril 2007.
- [http://www.logic-phone.com/Prod\\_01\\_01.htm](http://www.logic-phone.com/Prod_01_01.htm), visitado el 15 de abril de 2007.
- [http://www.entelcallcenter.cl/porque\\_elegirnos/nuestros\\_clientes.htm](http://www.entelcallcenter.cl/porque_elegirnos/nuestros_clientes.htm), visitado el 18 abril 2007.
- <http://reyporundia.phplibre.com.es/personal/apuntes/Apuntes%20de%20FIS.pdf>, visitado el 19 de abril de 2007.
- [http://www.usergioarboleda.edu.co/expopyme/aspectos\\_gastos\\_indirectos.htm](http://www.usergioarboleda.edu.co/expopyme/aspectos_gastos_indirectos.htm), visitado el 20 de abril de 2007.
- <http://www.gestiopolis.com/recursos3/docs/fin/macospvz.htm>, visitado el 25 de abril de 2007.