



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

**FACULTAD DE ESTUDIOS SUPERIORES
ARAGÓN**

**“SISTEMA PARA EL REGISTRO Y CONTROL DE FALLAS
TÉCNICAS DE LOS EQUIPOS DE RECARGA (POS)
INSTALADOS EN LAS TAQUILLAS DEL SISTEMA DE
TRANSPORTE COLECTIVO”.**

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

INGENIERO EN COMPUTACIÓN

P R E S E N T A N

VERÓNICA TAPIA GALICIA

HUMBERTO ROJAS VÉLEZ

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

SAN JUAN DE ARAGÓN ESTADO DE MÉXICO 2009





Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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



AGRADECIMIENTOS.



A mis padres por permanecer a mi lado y apoyarme en cada uno de los momentos de mi vida, por cimentar las bases de mi formación como persona y profesionalista; porque gracias a su ejemplo, esfuerzo, dedicación y apoyo esto ha sido posible. Sin duda son lo más importante en mi vida y esto es para ustedes, los Quiero.

A mis hermanos porque a su manera cada uno de ellos me ha apoyado y enseñado a lo largo de mi vida, deseando que esto sea un ejemplo para ustedes.

A mis amigos y compañeros por compartir conmigo momentos inolvidables, pero sobre todo sus conocimientos que sin duda forman parte de este proyecto.

A ti flakito por que de cualquier manera estuviste a mi lado hasta la culminación de este proyecto. Gracias por tu apoyo, Te Amo.

A la Universidad Nacional Autónoma de México por brindarnos la oportunidad de estudiar en la institución más importante de México.

A la FES Aragón por darnos los conocimientos para desarrollarnos como profesionistas pero ante todo como personas y buscar calidad en sus egresados.

VERÓNICA TAPIA GALICIA.



A mis padres quienes incondicionalmente me han dado todo su apoyo en cada uno de los aspectos de mi vida; a mi madre que me supo encaminar desde mis primeros pasos en la vida académica y que cuando fue rebasada en conocimientos siempre me apoyó y me alentó para poder seguir adelante; a mi padre que ha sido un ejemplo de superación y de tenacidad para ese hombre que no conoce los imposibles que día con día a forjado mi carácter para no rendirme ante nada que me enseñó que las cosas en la vida no son fáciles pero que con esfuerzo y dedicación se puede hacer cualquier cosa.

A mis hermanos que siempre han estado a mi lado y esperando que este paso les sirva de aliciente para ser y hacer más en la vida que cada uno de ellos lleva, por todas las cosas que hemos vivido y compartido les agradezco y los tengo presentes a cada momento con mucho cariño.

A todos y cada uno de mis amigos, a los que me ayudaron con la lectura y mejora de este trabajo, a quien me apoyo con su lap dejando de lado sus necesidades hasta donde le fue posible, a quienes me ayudaron en mis ratos de presión y me brindaron momentos de desahogo y sano esparcimiento, a quienes supieron escucharme en los momentos en que sentía que no terminaría y sin basilar me impulsaron para no dejar este proyecto en el olvido y con los que he compartido tantos años de experiencias buenas y malas.

A esa persona tan especial en este punto de mi vida que al verme perder mi tiempo en otras cosas me recordaba continuar mis avances y que representa tanto para mi y que sabe perfectamente que es ella.....

HUMBERTO ROJAS VÉLEZ.



ÍNDICE.



INTRODUCCIÓN	1
CAPÍTULO 1	5
ENTORNO DEL PROBLEMA	7
1.1 Antecedentes	7
1.2 Panorama General	11
1.3 Análisis de la Problemática	12
1.4 Causas más comunes de fracaso en el desarrollo de sistemas	17
1.4.1 Definición de Sistema	17
1.4.2 Etapas en el desarrollo de sistemas	18
1.4.3 Análisis de causas	19
1.5 Solución Propuesta	22
1.5.1 Arquitectura de software	22
1.5.2 Propuesta	23
1.6 Requerimientos de Hardware y Software	24
1.6.1 Software	24
1.6.2 Hardware	25
CAPÍTULO 2	29
MARCO TEÓRICO	31
2.1 Bases de datos relacionales y metodología de diseño	31
2.1.1 Introducción	31
2.1.2 Características	32
2.1.3 Cardinalidad	34
2.1.4 Álgebra relacional	36
2.1.5 Calculo relacional	38



2.1.6 Reglas de normalización.....	40
2.1.7 Metodologías de diseño.	42
2.2 Características, ventajas y desventajas de SQL Server 2005®.	47
2.2.1 Introducción.	47
2.2.2 Características de SQL.....	48
2.2.3 Ventajas de SQL Server 2005.....	52
2.2.4 Desventajas de SQL Server 2005.....	52
2.3 Características, ventajas y desventajas de Microsoft Visual Basic®.	53
2.3.1 Introducción.	53
2.3.2 Características.....	53
2.3.3 Ventajas.	60
2.3.4 Desventajas.	61
2.4 Características, ventajas y desventajas de Crystal Reports.	61
2.4.1 Características de Crystal Reports.....	61
2.4.2 Ventajas de Crystal Reports.....	66
2.4.3 Desventajas de Crystal Reports.....	68
2.5 Características, ventajas y desventajas de la Arquitectura Cliente - Servidor. 68	
2.5.1 Introducción.....	68
2.5.2 Características de la Arquitectura Cliente/Servidor.	70
2.5.3 Ventajas de la Arquitectura Cliente/Servidor.....	72
2.5.4 Desventajas de la Arquitectura Cliente/Servidor.....	73
CAPÍTULO 3	75
ANÁLISIS Y PLANTEAMIENTO DEL PROBLEMA.	77
3.1 Problemática Actual.	77
3.1.1 Problemas específicos al usar Excel.....	85



3.2 Requerimientos Generales y Particulares.....	86
3.2.1 Requerimientos Generales.....	86
3.2.2 Requerimientos Particulares.....	89
3.3 Identificación de los posibles módulos del sistema.....	91
3.4 Justificación en la elección de las herramientas de solución.....	94
3.4.1 El back-end.....	95
3.4.2 El Front-end.....	101
3.4.3 El Reporteador.....	104
CAPÍTULO 4.....	113
DISEÑO Y CONSTRUCCIÓN DEL SISTEMA.....	115
4.1 Elección de la metodología de desarrollo.....	115
4.2 Diagramación.....	116
4.2.1 Diagrama de Contexto.....	116
4.2.2 Diagrama de flujo de procesos.....	119
4.2.3 Diccionario de Datos.....	127
4.2.4 Diagrama Entidad Relación.....	135
4.2.5 Normalización.....	138
4.3 Diseño y construcción del back-end.....	144
4.4 Diseño y construcción del front-end.....	156
4.4.2 Módulo CAPTURA DE DATOS.....	160
4.4.3 Módulo CONSULTA DE PENDIENTES.....	163
4.4.4 Módulo CONSULTA DE ATENDIDOS.....	164
4.4.5 Módulo REPORTE.....	166
4.5.1. Pruebas del Sistema.....	168
4.5.1.2 Pruebas de Integración.....	170



4.5.1.3 Pruebas de Aceptación.	170
4.5.1.4. Otros tipos de pruebas.	171
4.5.2 Afinación de la Base de datos.	173
4.5.2.1 Afinación en la recuperación de datos.	174
4.5.2.2. Afinación respecto al uso del procesador.	174
4.5.2.3 Utilerías auxiliares.	175
4.5.3 Mantenimiento de la Base de Datos.	176
4.6 Obtención de Reportes.	178
CONCLUSIONES.	193
BIBLIOGRAFÍA.	197



INTRODUCCIÓN.



La necesidad de introducir la tecnología de acceso al Sistema de Transporte Colectivo por medio de Tarjeta sin contacto surge en primer lugar por la escasez o inexistencia de refacciones para mantener funcionando los torniquetes para el acceso por medio del tradicional boleto magnético de entrada unitaria y en segundo plano por la necesidad de enfrentar un problema ecológico a nivel mundial que hace necesaria la reducción en el consumo de papel además de que la fabricación del boleto resulta excesivamente caro para el STC.

Al integrar la Tarjeta de prepago al esquema de peaje en el Sistema de Transporte Colectivo, es necesario contar con los medios necesarios para “recargar” estas tarjetas, estos medios son básicamente dos, el primero es por medio de maquinas expendedoras de recarga automática que no dan cambio al usuario, es decir, recargan el monto recibido y el segundo en las taquillas del STC recargando la cantidad especificada por el usuario y con la posibilidad de obtener cambio.

Actualmente en el STC se encuentran instalados 332 equipos POS a lo largo de las 175 estaciones que componen las 11 líneas de la Red del metro, los cuales tienen como objetivo la recarga de 1,766,128 tarjetas recargables que se encuentran en uso.

El crecimiento de la demanda para la utilización de la Tarjeta recargable hace que día a día cada POS instalado tenga un mayor número de tarjetas a recargar, lo que lleva a un desgaste técnico de los equipos.

Con la finalidad, de dar un servicio óptimo y eficaz a los usuarios del STC, un POS no puede permanecer (en un panorama ideal) apagado o fuera de servicio un tiempo mayor a 3 horas; por lo que es necesario contar con las herramientas y refacciones necesarias para estar en condiciones de reparar y/o cambiar un equipo en el menor tiempo posible.

Dado lo anterior, es necesario contar con información de calidad y consistente para poder adquirir y justificar mes a mes las refacciones necesarias, así como



poder asignar al personal de esta área los estímulos monetarios correspondientes, por su desempeño.

Es entonces que se hace necesario contar con información organizada y disponible en todo momento ya que por medio de esta, el área de Soporte Técnico al POS conocerá la(s) refacción(es) que se averían con mayor frecuencia en dichos equipos, además de realizar el análisis adecuado para compras futuras.



CAPÍTULO 1



ENTORNO DEL PROBLEMA.

1.1 Antecedentes.

La implantación de Tarjeta Inteligente como tecnología única para el control de peaje en estaciones es un proyecto de modernización que se ha venido implementando en el Sistema de Transporte Colectivo (STC).

El esquema de peaje que opera en el STC se integra por dos tecnologías: boletos con banda magnética y tarjeta inteligente.

El 99% del control del peaje se realiza mediante tecnología de tratamiento magnético lo que impone un alto costo de operación y mantenimiento, además de estar expuestos a los riesgos inherentes a la logística del pago con boletos.

A continuación se mencionan las principales condicionales:

- Los equipos de peaje han rebasado su vida útil.
- El mantenimiento al equipo actual es costoso.
- La producción y logística del manejo del boleto son costosas.
- El control del esquema se realiza manualmente.
- El esquema de pago por boleto magnético es propenso a fraudes.
- El esquema de peaje no proporciona información útil para el mejoramiento del servicio.
- El sistema actual de peaje no permite el monitoreo de las transacciones realizadas.

En la figura 1.1.1 se detalla el esquema general del funcionamiento con boleto magnético:

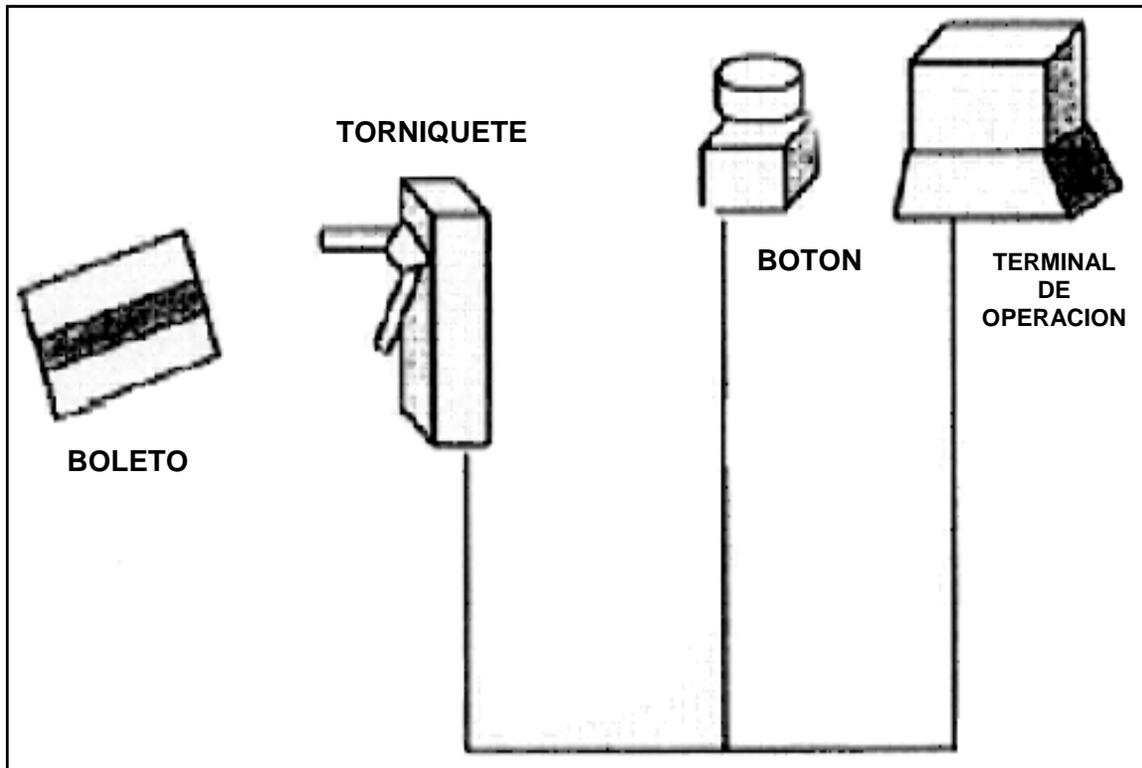


Figura 1.1.1 Esquema general funcionamiento con boleto magnético.

- El boleto de cinta magnética contiene la información necesaria para la validación del peaje de un solo viaje. El usuario adquiere, únicamente en las taquillas, los boletos unitarios necesarios e introduce uno por cada viaje que realiza.
- El torniquete de entrada, en un proceso completamente local y autónomo, recibe el boleto, lee la cinta magnética y en su caso concede o niega el pase, acción informada al mecanismo mediante una señal eléctrica y al usuario mediante señales luminosas en el pictograma frontal y en la taquilla de salida en la parte superior.
- Para el otorgamiento de entradas de cortesía, inicialmente un botón instalado en taquilla y luego en el torniquete directamente, permite la liberación del trípode.
- La terminal de operación, instalada en el cubículo del jefe de estación, permite el control y estadística de los accesos de una estación en particular.

Como puede verse, la solución es local y no permite la concentración de la información de afluencia, por lo que los datos referentes al acceso deben tomarse directamente torniquete por torniquete.

Uno de los principales objetivos de la implementación de la Tarjeta inteligente es el concentrar la información de afluencia y obtener resultados que incidan en la operación y planeación así como también relacionar los datos de venta y recarga con los pasajes otorgados.

Es por ello que el STC, ha implementado el uso de tarjeta inteligente, utilizando el siguiente esquema por estación. Vea figura 1.1.2:

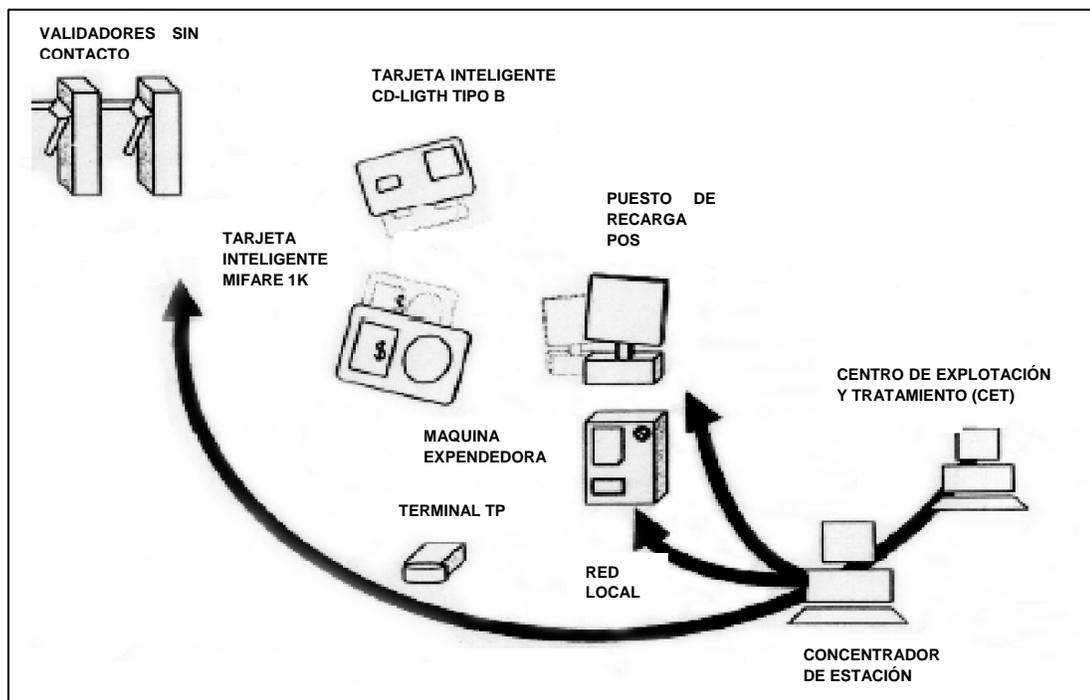


Figura 1.1.2 Esquema por estación.

- **Validadores sin Contacto:** se encuentran instalados en cada una de las estaciones, y es el equipo responsable de otorgar o negar los pases con tarjeta inteligente.



- **Tarjeta inteligente CD-Light tipo B:** es la tarjeta utilizada para otorgar pases de cortesía a trabajadores, derechohabientes y supervisores del STC Metro, así como a las personas de la tercera edad y discapacitados.
- **Tarjeta inteligente Mifare 1K:** es la tarjeta utilizada para el prepago y pago de los viajes realizados por los usuarios.
- **Terminal TP:** Es la terminal con la que a través de infrarrojo permite la recolección de los archivos de actividad generados por los validadores, así como actualizar los parámetros de los mismos.
- **Punto de Recarga (POS):** Se encuentra instalado en cada una de las taquillas del STC y es el equipo encargado de realizar las recargas a las tarjetas de prepago (Mifare 1k), de acuerdo a lo solicitado por el usuario, así como la verificación del saldo de la misma.
- **Máquina Expendedora:** Equipo en el que se puede recargar y/o comprar una tarjeta de prepago (Mifare 1K), así como la verificación del saldo de la misma.
- **Red Local:** Es la conexión entre equipos mediante cable FTP y fibra óptica que permite que todos los archivos de actividad generados en cada uno de los equipos (validaciones, recargas, compras) se han enviados al Concentrador de estación.
- **Concentrador de Estación:** Es el equipo encargado de concentrar toda la información de actividad generada en una estación.
- **Centro de Explotación y Tratamiento (CET):** Es el equipo el cual se encuentra instalado en las instalaciones del STC, el cual almacena toda la información de actividad generada por cada una de las estaciones, la recibe a través de los concentradores de estación.

Como se puede observar con este nuevo sistema es posible la concentración de la información de afluencia, venta y recargas.



Para la realización de este proyecto de tesis, nosotros nos enfocaremos a las fallas técnicas presentadas y registradas por los Puntos de Recarga (POS), ya que con el incremento de las tarjetas inteligentes de prepago el desgaste de dichos equipos se ha incrementado.

1.2 Panorama General.

En cada taquilla de la red del STC existe un equipo necesario para la recarga de las tarjetas inteligentes sin contacto. Este equipo se encuentra instalado en el pupitre del personal de taquilla.

El personal de taquilla utiliza dicho equipo para realizar todas las operaciones de recarga de la tarjeta de prepago a los usuarios, y efectúa el registro de las transacciones llevadas a cabo durante su turno, generando un histórico de la información, todo esto con la previa digitación de un password de acceso asociado al operador.

El equipo de recarga POS se encuentra integrado por los siguientes elementos:

- 1. Cuerpo del CPU:** El gabinete da soporte a todos los subconjuntos mecánicos, eléctricos y electrónicos internos, al igual que a las unidades externas de comunicación usuario/equipo, es robusto, con ventilación adecuada y filtros que evitan la entrada de polvo.
- 2. Un teclado para la introducción de datos, acceso al mantenimiento y operación del equipo:** Es un teclado alfanumérico, a través del cual el personal de operación y mantenimiento pueda establecer comunicación con el equipo.
- 3. Un visualizador de información al usuario y operador:** Contienen una pantalla por medio de la cual se presenta toda la información necesaria para el usuario y personal de operación y mantenimiento.
- 4. Conjunto electrónico:** Está formado por el procesador central y sus circuitos asociados para censar variables y activar elementos de salida, todos estos



están concentrados en un solo compartimiento. Esta circuitería está organizada de una manera lógica y modular en tarjetas de circuito impreso, provistas de los conectores necesarios para su enlace con el resto de los subconjuntos y dispositivos eléctricos del equipo, evitando conexiones erróneas.

5. **Interfaces de comunicación:** Para la comunicación al exterior del equipo, este contiene un puerto que permite la comunicación con una computadora de estación (RS485, USB y Ethernet).
6. **Fuente de alimentación protegida, con interruptor de encendido/apagado y UPS:** La alimentación de entrada de los equipos es automáticamente configurable a 110 VCA/220 VCA a 60 Hz. La alimentación de salida VCD para los circuitos electrónicos del equipo, se suministra a partir de una fuente de alimentación diseñada como un bloque modular en donde se generan las diferentes tensiones requeridas, y cuya salida está dada a través de varios conectores identificando la alimentación suministrada a cada uno de ellos.
7. **Impresora de agujas:** Cuenta con una impresora, con la finalidad de emitir tanto la ficha de corte de caja, es decir; el total de recargas por turno, así como el comprobante de recarga al usuario.
8. **Módulo lector (CSC):** Es el equipo electrónico que contiene la información eléctrica que debe contener cada una de las tarjetas de prepago, para su correcta lectura. En dicho equipo, es colocada la tarjeta para poder realizar la verificación de saldo así como las recargas a la misma.

1.3 Análisis de la Problemática.

Actualmente se encuentran instalados 332 equipos POS con las características descritas en el capítulo 1.2, los cuales tienen como objetivo la recarga de 1,766.128 tarjetas recargables que se encuentran hoy en día en uso.

Dado al incremento en los últimos meses de la venta de tarjeta recargable en el STC, y a la utilización de estos equipos, se ha comprobado el desgaste técnico de algunos elementos que los integran.



Así mismo y dado a que los equipos han sido instalados y destinados a ser utilizados por personal que cuenta con un perfil diferente al técnico, el uso de los mismos en algunos casos no ha sido el óptimo, pues se han presentado casos donde las fallas y/o problemas técnicos han sido provocados por causa humana, adicionalmente otro de los problemas identificados es que no se cuenta con el material necesario para realizar la óptima reparación de los equipos, debido a que siendo un problema técnico no se tiene el inventario necesario para saber si la pieza que se va a reemplazar está vigente, o bien; es necesario realizar una solicitud de compra, lo cual implica retrasos considerables en los tiempos de respuesta después de haberse presentado una falla en los equipos.

Por tal motivo se creó el área de Soporte Técnico al POS, la cual pertenece a la **GOS** (Gerencia de Organización y Sistemas), y tiene como objetivo el registro, control y reparación de cada una de las fallas técnicas presentadas en estos equipos.

Esta área cuenta con un total de 11 personas, de las cuales 6 tienen calidad laboral eventual, y deben estar comprobando mes a mes su desempeño laboral para así poder ser acreedores a estímulos monetarios.

Debido a que el objetivo del STC con sus usuarios es brindarles un servicio óptimo y eficaz, el organismo debe contar con las refacciones y/o elementos necesarios para poder reparar un equipo POS en el menor tiempo posible.

Hoy en día el proceso de reparación para un POS es bastante tardado y burocrático ya que actualmente el registro y control de dichas fallas se lleva a cabo de manera manual, lo que causa un atraso en los informes y reportes estadísticos de dichas fallas, esto debido a que no se tiene una herramienta o sistema que permita tener un control adecuado y eficaz sobre las fallas más frecuentes que hoy en día tienen los POS, por tal motivo es indispensable la creación de un sistema que permita al personal de Soporte Técnico capturar la información relacionada



con los reportes de fallas en los POS, ya que al tener los reportes de forma manual como se lleva a cabo actualmente, la reparación de un POS puede tardar desde un día hasta semanas por no tener un dato correcto sobre la falla reportada;, esto ocasiona grandes pérdidas al STC al no poder efectuar las transacciones correspondientes en los equipos de recarga cuando son requeridos.

Por otro lado, el no contar con la información adecuada en tiempo de los reportes de fallas registrados y atendidos por cada una de las personas del área de soporte técnico al POS, provoca un retraso en el pago de estímulos monetarios por desempeño, ocasionando inconformidad y constantemente el cambio del personal de Soporte Técnico, lo cual implica pérdidas de tiempo y duplicidad de trabajo en los trabajadores permanentes, al tener la necesidad de capacitar a las personas que ingresan al área.

Otro problema ocasionado por no tener la información veraz y actualizada en tiempo y forma es la adquisición de las refacciones y/o elementos necesarios para la reparación de un POS debido a que no se cuenta con estadísticas sobre las principales fallas reportadas que permitan al área de soporte identificar las causas del problema con mayor eficacia y con ello ofrecer un mejor servicio al momento de realizar una reparación.

Actualmente el proceso para la reparación de un POS inicia cuando el usuario (usualmente el área de taquillas), detecta algún problema con los equipos POS, este es reportado vía telefónica al área de soporte técnico, el responsable en turno, atiende la falla asignándole un número consecutivo para identificarla, tratando de dar solución por el mismo medio.

Si la falla es resuelta vía telefónica, el personal de soporte técnico registra la falla y solución que le brindó; de lo contrario, se asigna a un ingeniero para llevar a cabo en sitio la revisión del equipo, si se cuenta con la refacción y/o material indicado y requerido para resolver el problema presentado, el ingeniero



responsable resuelve la falla en sitio; de no ser así se deberá llevar a cabo un trámite burocrático el cual se lleva aproximadamente 10 días hábiles, que consiste en que el área de soporte técnico al POS deberá solicitar a la GOS la pieza, refacción y/o material necesario para resolver la falla.

Una vez que se ha hecho la solicitud a la GOS, esta valida la existencia de la pieza, refacción y/o material solicitado en bodega, si existe; la GOS envía lo necesario al área de soporte técnico al POS para resolver la falla en sitio.

Sin embargo, al no tener el registro óptimo de la pieza (s), refacción (es) y/o material (es) que con mayor frecuencia requiere de ser atendido y/o cambiado por parte del área de soporte técnico, en la mayoría de los casos la GOS no cuenta con el material indicado y necesario para resolver al momento y con mayor rapidez la falla presentada, por lo que esa Gerencia deberá hacer una solicitud al área de Adquisiciones del STC para que ésta realice la compra y/o autorice el fondo monetario necesario para la adquisición de dicho material.

Teniendo la GOS la pieza (s), refacción (es) y/o material (es) necesario para resolver la falla, este es enviado al área de soporte técnico, para llevar a cabo la solución de la falla.

En el diagrama 1.3.1 se muestra el proceso que actualmente se lleva a cabo para la solución de una falla en un equipo POS.

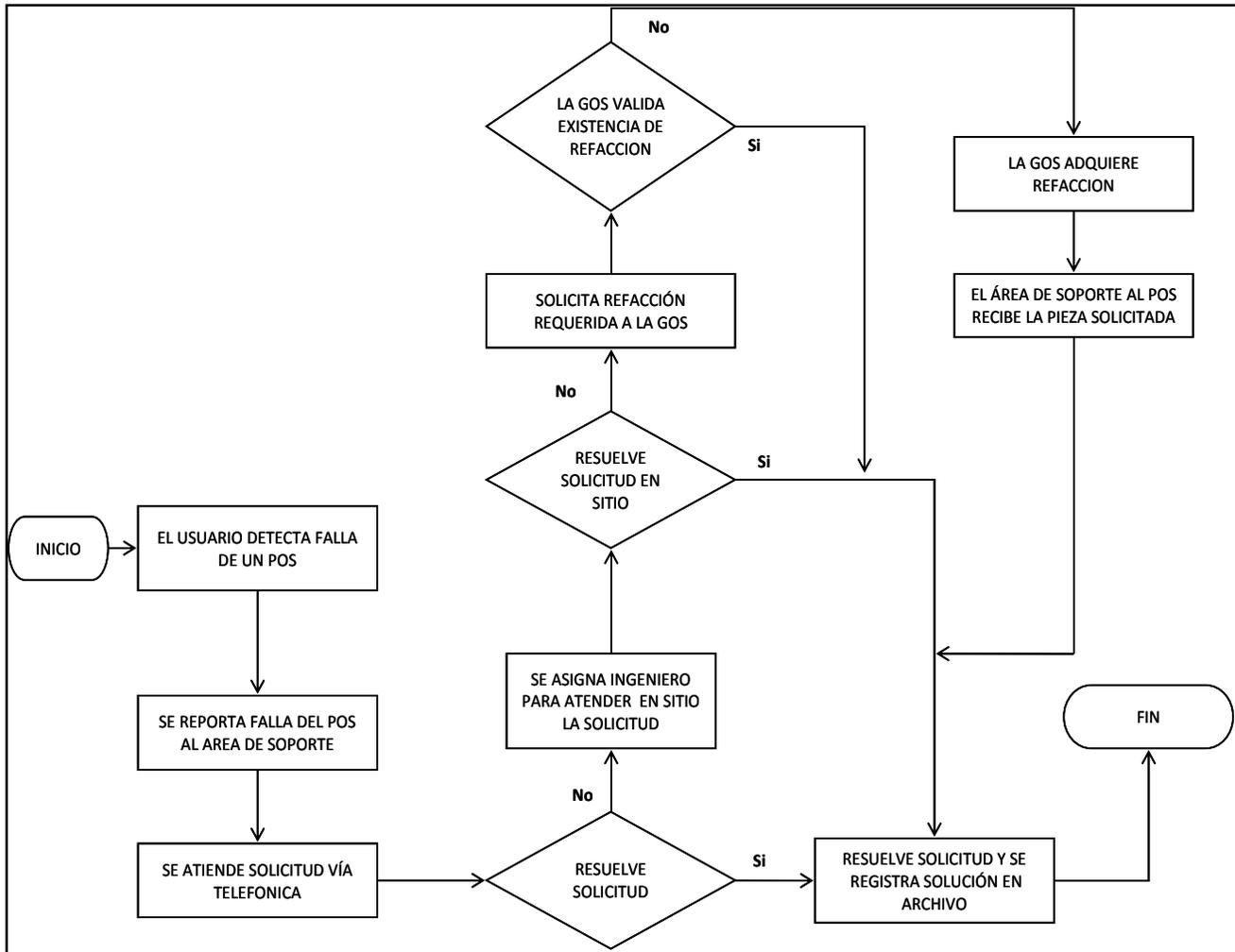


Figura 1.3.1 Diagrama proceso para la solución de una falla en un equipo POS.

Por lo anterior, el diseño de un Sistema para el registro y control de fallas técnicas de los equipos de recarga (POS) instalados en las taquillas del Sistema de Transporte Colectivo, tiene como objetivo:

- Tener un formato estándar (pantalla) para la captura de fallas reportadas y con esto cualquier personal del área pueda dar seguimiento a una solicitud.
- Tener reportes que muestren una relación de las fallas más frecuentes en los equipos.



- Tener reportes que muestren la (s) refacción (es), que presenten con mayor frecuencia una falla, para poder contar con un stock en bodega, y así disminuir el tiempo en la adquisición de las mismas.
- Generación de estadísticas y reportes sobre los equipos con mayor reincidencia de fallas, esto con la finalidad de hacer un análisis más detallado e identificar posibles soluciones de fondo.
- Cada reporte deberá tener los datos requeridos como: Usuario, POS, fecha, falla reportada, taquilla y estación pertenecientes para poder brindar el soporte adecuado en el menor tiempo posible.
- Tener un catalogo de fallas frecuentes que permitan al personal de soporte identificar de forma optima y rápida su posible solución al tener una descripción general del problema.

1.4 Causas más comunes de fracaso en el desarrollo de sistemas.

1.4.1 Definición de Sistema.

En general un sistema puede ser definido como un conjunto de componentes que interactúa entre si con el fin de alcanzar un objetivo común, y pueden ser representados en su mayoría por el siguiente diagrama mostrado en la figura 1.4.1:

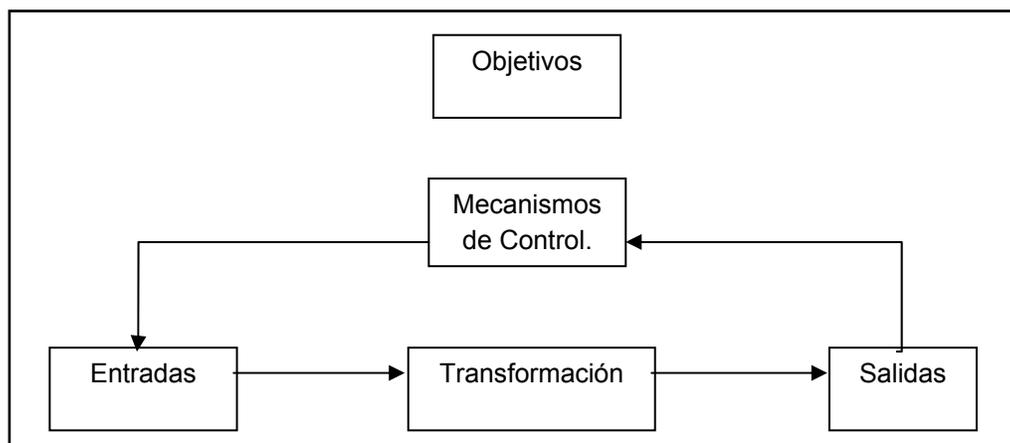


Figura 1.4.1.1 Representación general del flujo de información en un sistema.



Desde el punto de vista informático un sistema se puede definir como el conjunto de elementos tales como: programas (software), procedimientos, datos y equipo (hardware) utilizado en el procesamiento de información, por ejemplo: un sistema de contabilidad, un sistema de facturación o un sistema de gestión de base de datos, que interactúan entre sí con el fin de apoyar las actividades de una actividad en particular dentro de un negocio, una empresa o como en este caso el sector de servicios públicos.

Un sistema de información realiza a nivel interno cuatro actividades básicas sobre los datos o información que recibe y estos son: entrada, almacenamiento, procesamiento y salida de dicha información.

1.4.2 Etapas en el desarrollo de sistemas.

Para desarrollar una aplicación o sistema se requiere utilizar metodologías y modelos ya que es la mejor forma de alcanzar los objetivos planteados y de una forma más eficiente.

A grandes rasgos, los pasos para desarrollar un nuevo sistema de información son:

- Definir los requerimientos y prioridades de la empresa o institución.
- Cuantificar el problema a ser resuelto.
- Identificar los principales componentes de la solución.
- Hacer un esbozo general que describa las relaciones entre estos componentes.
- Agregar cualidades propias del sistema, por ejemplo: seguridad, escalabilidad, disponibilidad, etc., y los niveles de dependencia entre el sistema y el ambiente de operación.
- Análisis de los componentes y desarrollo del sistema.
- Pruebas, documentación y finalmente mantenimiento del sistema.

Lo que se busca es proyectar de la mejor forma posible lo que el cliente desea, esto según la percepción del desarrollador. En algunos casos lo que el usuario desea no es viable o representaría un alto costo y esto es necesario aclararlo desde el principio.

1.4.3 Análisis de causas.

Los problemas que ocasionan que un sistema de información falle total o parcialmente caen en diferentes categorías como se ilustra en la figura 1.4.3.1.

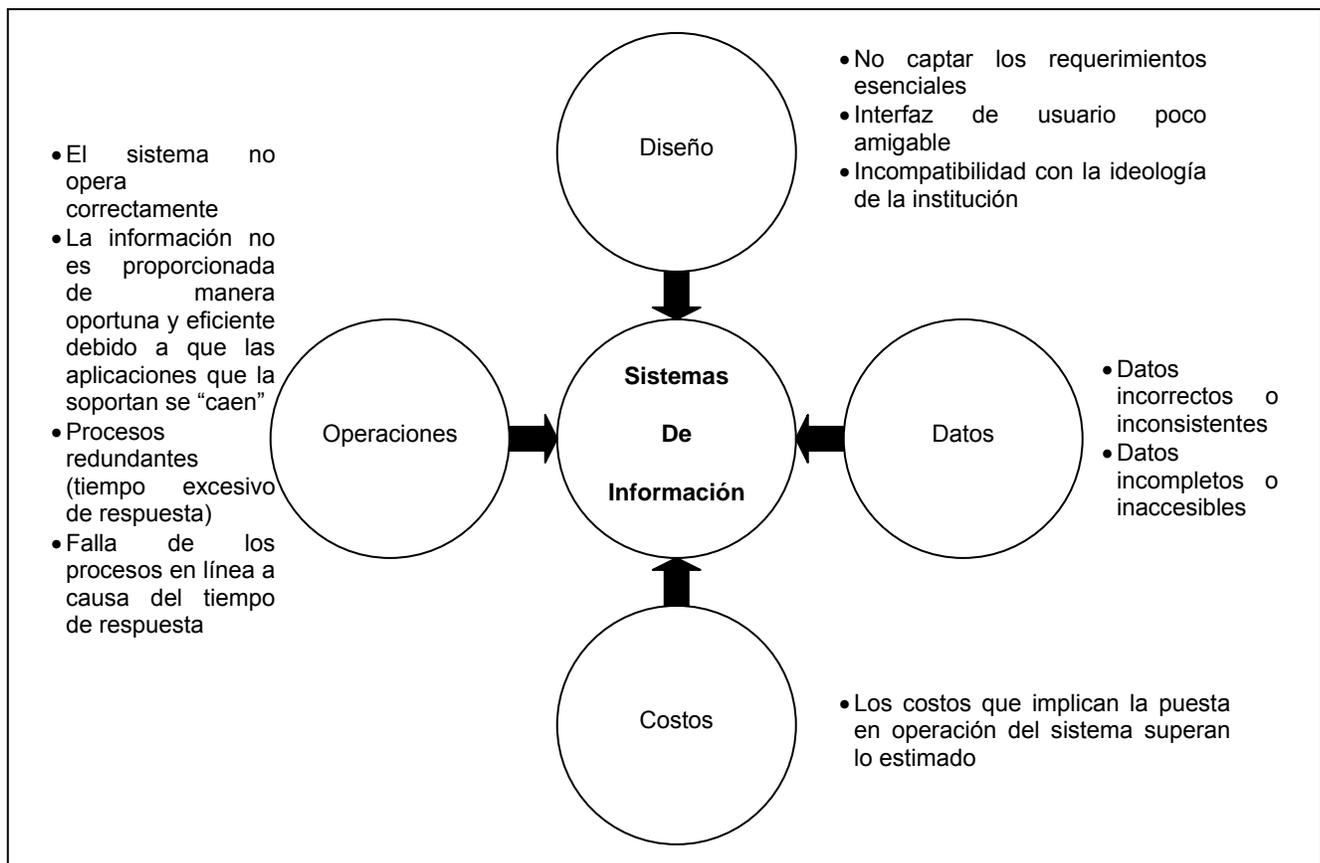


Figura 1.4.3.1 Representación gráfica de las posibles causas de fracaso en el desarrollo e implantación de un sistema.

DISEÑO: El diseño de un sistema falla si no son entendidos los requerimientos reales y esenciales del cliente para el que está dirigido. La información puede no ser proporcionada lo suficientemente rápido para ser útil; puede venir en un formato imposible de entender y por lo tanto de usar o puede



representar elementos equivocados de datos. La manera en la cual usuarios no técnicos del nuevo sistema deben interactuar con éste, puede ser excesivamente complicada y desalentadora, es en este sentido que adquiere mucha importancia el buen diseño de la interfaz gráfica de usuario (GUI), contemplar el perfil del personal que operará el sistema es fundamental para el éxito de éste.

En este caso lo más común es que no se haya recabado la información adecuada acerca del problema en la aplicación de entrevistas, cuestionarios, etc., por lo que su posterior análisis dará resultados equivocados.

DATOS: Los datos en el sistema tienen un alto grado de imprecisión o de inconsistencia. La información en ciertos campos puede ser errónea o ambigua: puede no ser fragmentada adecuadamente para resolver el problema que pretende. La información que se requiere para una función específica puede ser inaccesible porque los datos están incompletos o por el empleo de un control excesivo e inadecuado de bloqueos o contraseñas.

COSTOS: Debemos tener en cuenta que implantar el sistema en muchas ocasiones implica realizar un cambio a nivel técnico (capacitación), y en otras pocas, cambios a nivel organizacional, y que casi siempre pasan inadvertidos los problemas significativos del cambio organizacional, de ahí el fracaso de la implantación técnica.

El costo de la implantación puede ser alto en comparación con el diseño, la razón de ello es el elevado precio de los paquetes de software (aunque una alternativa es el software libre), la capacitación, el equipo y suministros, la depuración y preparación de los manuales de operación (documentación).

Es recomendable para el desarrollador proporcionar varias opciones de solución con un costo/beneficio viable y además, hacer recomendaciones acerca de lo más conveniente para quien adquiere el sistema.



OPERACIONES: El sistema no opera bien, es decir, la información no se proporciona de manera oportuna y eficiente porque uno o varios de los procesos computacionales que intervienen en el procesamiento de la información se “caen”. Las herramientas elegidas tal vez no son las más adecuadas o simplemente los estudios de factibilidad no fueron evaluados por todas las personas involucradas en el proceso.

Es importante que el sistema funcione de forma rápida, que los resultados sean mostrados en una interfaz (formato) amigable para el usuario final, pero a la vez que sea confiable y esto se lograra desarrollando un código eficiente, es decir evitar los ciclos y los bloques que no sean necesarios para cumplir con el objetivo del sistema.

¿Qué puede fallar?

- Se presupuestan tiempo y dinero insuficientes para las actividades de conversión, en especial conversión de datos.
- No todas las personas que pudieran utilizar el sistema están involucradas, hasta que da inicio la conversión o implantación. Algunos ejecutivos generales y funcionales creen que el desarrollo de sistemas es un tema estrictamente técnico que sólo debe ser del interés de las personas técnicas, pero esto en realidad no es así. El personal operativo debe participar en el proceso de desarrollo y debe entender todas las fases. Es necesario que participen también en las decisiones relativas a la construcción del sistema y en las de selección de software.
- La capacitación del personal involucrado inicia cuando el sistema esta a punto de ser instalado, o en el peor de los casos cuando ya esta en operación.
- El sistema es puesto en operación antes de estar al 100% con el objetivo de recuperar costos o tiempos.
- Documentación del sistema y para el usuario inadecuados.
- No se considera el mantenimiento o se implementa de manera inadecuada.



1.5 Solución Propuesta.

1.5.1 Arquitectura de software.

Para lograr como resultado un Sistema de Información con una mayor probabilidad de satisfacer las necesidades del usuario final y que por tanto resuelva de un modo más eficiente el problema para el que es diseñado, es necesario usar métodos tales como la Arquitectura de Software (o Arquitectura Lógica).

La Arquitectura de Software esta conformada por un conjunto de pasos y definiciones que hacen posible la obtención de un marco de referencia que servirá como guía en la construcción del software para un Sistema de Información y establece las bases para que analistas, diseñadores, programadores y en general las personas involucradas en el desarrollo del Sistema, trabajen en una línea común y bien definida, con el fin de alcanzar los objetivos que han sido planteados por el mismo.

Las arquitecturas de software más conocidas son las siguientes:

- **Monolítica:** El software está conformado por grupos funcionales perfectamente acoplados entre sí.
- **Cliente-Servidor:** Esta arquitectura Sustituye a la monolítica consiste básicamente en un cliente que realiza peticiones a otro programa que le da respuesta y que se conoce como servidor (permitiendo repartir la carga de trabajo a nivel lógico y físico), la especialización en la división de tareas en esta arquitectura es mencionada por algunos autores como Arquitectura de tres niveles, pero otros tantos la conciben como un perfeccionamiento de la Arquitectura Cliente - Servidor cabe mencionar que en otro apartado del presente trabajo ampliaremos más sobre este tema.

La Arquitectura de Software se selecciona y diseña con base en los objetivos y restricciones del Sistema, aunque es necesario considerar a la vez las ventajas y desventajas de la Arquitectura en si.

La arquitectura Cliente – Servidor puede ser representada por el siguiente diagrama de la figura 1.5.1.1.

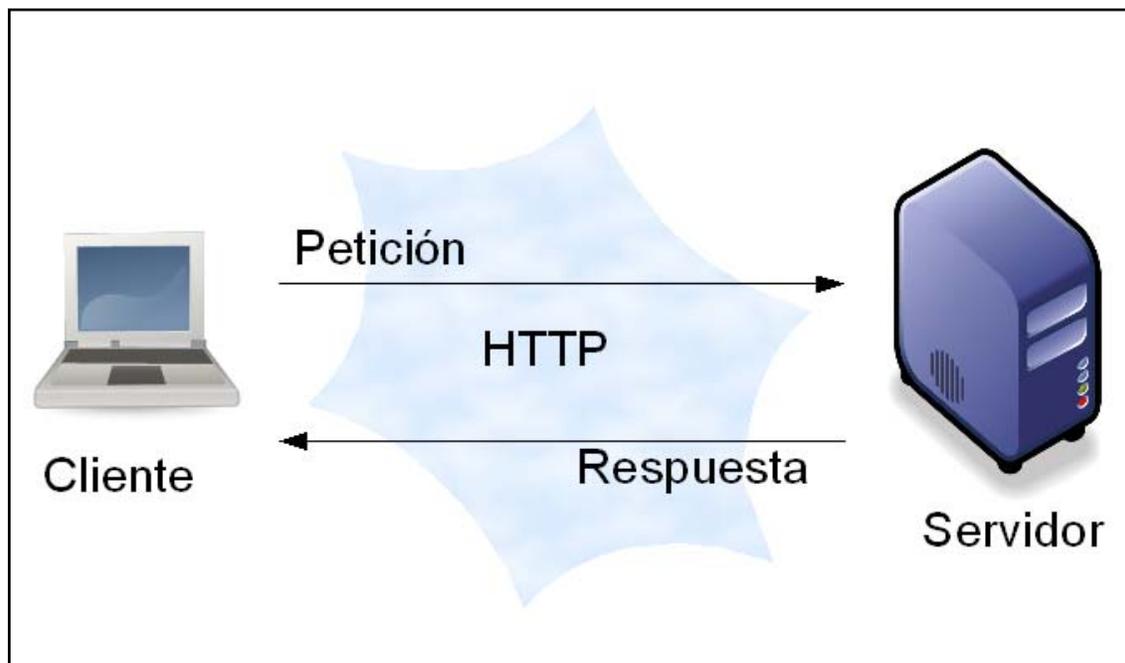


Figura 1.5.1.1 Representación Gráfica del la Arquitectura Cliente – Servidor.

1.5.2 Propuesta.

En base a lo expuesto anteriormente, se propone desarrollar un sistema bajo la Arquitectura Cliente – Servidor bajo el patrón MVC (Modelo Vista Controlador), en el cual el Sistema queda conformado básicamente por tres componentes, que son la interfaz de usuario (vistas), la lógica de control (controlador y modelo) y los datos de aplicación (La base de datos).

Esta separación de funciones, nos permite tener un Sistema con una mayor facilidad de mantenimiento y modificación en el largo plazo.



Modelo: Es una forma de llamar a la capa de dominio, la lógica de dominio es la que dota de significado a los datos, por ejemplo es tiempo transcurrido entre la fecha en que se reporta una falla y la fecha en la que se le proporciona atención.

Vista: Es la manera de presentar el modelo en un formato adecuado para interactuar de manera cotidiana con el usuario final, es decir, una interfaz de usuario.

Controlador: En general, este responde a eventos o acciones del usuario (consultas o inserción de datos) y provoca cambios en el modelo, en la vista y probablemente en la Base de Datos.

Particularmente, nuestra propuesta consiste en:

- Una Base de Datos bajo el modelo relacional desarrollada en Microsoft SQL Server 2005 ®.
- El controlador estará implementado desde el DBMS proporcionado por Microsoft SQL Server 2005 ®.
- Una interfaz de captura para la base de datos implementada con Microsoft Visual Basic 6.0 ®.
- Y una interfaz para la implementación de reportes Desarrollada en Crystal Reports ®.

1.6 Requerimientos de Hardware y Software.

1.6.1 Software.

Los requerimientos de Software para la implementación del Sistema de Información propuesto con los siguientes:

Microsoft Windows XP Professional con Service Pack 2®, es el Sistema Operativo que se encuentra instalado en el área de Soporte Técnico al POS del Sistema de Transporte Colectivo y que además cumple con las características subyacentes



del resto de los programas que serán utilizados y se cuenta con las licencias necesarias para su uso.

Microsoft SQL Server 2005 ®, La razón por la cual se sugiere el Structured Query Language o SQL (**Lenguaje Estructurado de Consultas**), es debido a que es una herramienta útil para organizar, gestionar y recuperar datos almacenados en una base de datos, es un lenguaje informático estandarizado que se puede utilizar para interactuar con una base de datos y en nuestro caso específico con un tipo conocido como base de datos relacional y que nos permite realizar todas las operaciones que se pueden hacer en una base de datos; además de la disponibilidad de licencias por parte del Sistema de Transporte Colectivo.

Microsoft Visual Basic 6.0 ®, el diseño de la interfaz de captura o ingreso de datos a la base sugerimos sea implementada a través de esta herramienta de desarrollo, esto debido a la compatibilidad que este programa y las aplicaciones desarrolladas con el tienen con el entorno Windows XP Professional con Service Pack 2 ®, que es el Sistema Operativo que se encuentra funcionando en el área de Soporte Técnico al POS del Sistema de Transporte Colectivo, y a la facilidad que este brinda para la implementación de interfaces gráficas que a la vez sean amigables para el usuario final de dicha interfaz.

Crystal Reports ®, Finalmente, la utilización de esta herramienta de explotación para la Base de Datos, con el objetivo de hacer más fácil la obtención de información concreta de la base para la elaboración de reportes específicos, estadísticas de productividad, de tiempos de atención, taquillas con mayor incidencia, fallas más comunes, etc., en tablas y de manera gráfica.

1.6.2 Hardware.

La mayor demanda de recursos en cuanto a Hardware, la tiene el sistema operativo, en el caso de Microsoft Windows XP Professional con Service Pack 2®, los requerimientos mínimos son:



- **Procesador:** Intel Pentium a 233 MHz o Celeron a la misma velocidad; de Athlon el K6 o Duron en AMD, aunque el recomendado es PIII a partir de 500 MHz.
- **Memoria RAM:** Mínimo 128 Mb en RAM, aunque la recomendación es de 512 Mb para que el sistema operativo sea estable.
- **Disco Duro:** 3 Gb como mínimo, aunque esto es solo de Sistema Operativo, en realidad esto si depende en gran medida de la cantidad de aplicaciones que serán instaladas en el equipo.
- **Tarjeta de Vídeo:** SVGA a 800x600 con 32 MB y con soporte 3D Directx 8.
- **Monitor:** Con soporte de resolución mínimo 800x600, recomendado 1024x768.
- Por supuesto Teclado, Ratón y Unidad Lectora de CD ROM o DVD.

Para Microsoft SQL Server 2005 ®:

- De inicio requiere de Microsoft Windows XP Professional® en cualquiera de sus versiones, para la edición Developer Edition y Express Edition.
- Memoria RAM: Por lo menos 512 Mb, se recomienda 1 Gb.
- Disco Duro: Requiere de un espacio libre de por lo menos 390 Mb (Para la instalación solamente).
- Unidad Lectora de CD ROM o DVD.
- Microsoft Internet Explorer Versión 5 o posterior.

En el caso de Microsoft Visual Basic 6.0 ®:

- Windows 98, Windows 2000, Windows XP® con un Personal Web Server instalado o con Internet Information Server 4.0 o superior.
- Procesador: Mínimo Pentium III a 500 MHz o compatible.
- Memoria RAM: Mínimo 256 Mb en RAM.
- Disco Duro: Mínimo 100 Mb libres (Solo para la instalación).
- Unidad Lectora de CD ROM o DVD.
- Monitor: Mínimo VGA, se recomienda SVGA.
- Mouse u otro dispositivo puntero.



Y finalmente para Crystal Reports ®:

- Sistema Operativo: Windows 2000®, Windows XP Professional®, Windows 2003 Server ®.
- Procesador: Mínimo Pentium III a 500 MHz.
- Memoria RAM: Mínimo 128 Mb, se recomienda 512 Mb.
- Disco Duro: 320 Mb libres como mínimo (para instalación).
- Unidad lectora de CD ROM o DVD.





CAPÍTULO 2



MARCO TEÓRICO.

2.1 Bases de datos relacionales y metodología de diseño.

2.1.1 Introducción.

Las bases de datos son esenciales para un sistema de información, este tiene como función mantener los datos para éstas y auxiliar a los usuarios a interpretar los datos para tomar decisiones. El papel de la Base de Datos es fundamental ya que es el depósito de los datos en el Sistema de Información.

Los usuarios que deben tomar decisiones obtienen los datos necesarios al acceder a la base y ejecutar las consultas adecuadas en ella. La localización de la Base de Datos y la facilidad para acceder a ella están directamente relacionadas con el grado de efectividad del Sistema de Información. La facilidad en el acceso a una variedad de datos desde un número definido de lugares permite que dicho Sistema tenga buena capacidad para responder a las necesidades de quienes toman decisiones relacionadas con los resultados, que éste proporcione, y por el contrario, si se tiene un acceso deficiente a los datos, se dificulta una rápida respuesta debido a que no se dispone con facilidad de los mismos; esto dará como resultado que las decisiones se retarden en forma innecesaria o sean tomadas basándose en datos incompletos, lo que puede conducir a un mal desempeño posterior del sistema.

Para poder entender mejor el lenguaje utilizado en las Bases de Datos, es necesario definir algunos conceptos:

DATO: Es la unidad mínima de información, una representación simbólica (numérica, alfabética, etc.) que define el valor de un atributo o característica de una entidad. El dato no tiene significado en sí mismo, pero al ser procesado de manera adecuada, se puede utilizar en la realización de cálculos o toma de decisiones.



INFORMACIÓN: Es un conjunto organizado e interrelacionado de datos, producto de un procesamiento adecuado de los mismos. La información sí posee significado y nos permite obtener conocimiento para una futura toma de decisiones.

ANÁLISIS DE LA INFORMACIÓN: Es la representación conceptual de la problemática presente y que se desea resolver, y que tiene como característica principal la claridad en su contenido.

BASE DE DATOS: Es un conjunto de datos que se encuentran relacionados entre sí con un objetivo común. También puede ser definida como una colección de datos integrados, que tienen una redundancia controlada y que poseen una estructura que refleja las interrelaciones y las restricciones que existen en el mundo real.

2.1.2 Características.

Los datos existentes en una base, deben ser independientes de los diferentes tipos de usuarios y aplicaciones que los compartirán, es por esto que la definición y descripción que se hace para cada tipo de dato debe ser almacenada junto con estos y los procesos de actualización y recuperación comunes entre usuarios y aplicaciones deben garantizar la integridad, seguridad y confidencialidad del conjunto de datos.

Los tres componentes principales de un sistema de bases de datos son: el hardware, el software y los datos a manejar; adicionalmente debe ser considerado el personal encargado del sistema.

Las principales características de una Base de Datos son:

No redundante: Este aspecto hace referencia a que no exista información repetida o duplicada dentro de las diferentes tablas de la Base de Datos.



Consistente: La inconsistencia de datos por lo general se debe a problemas de redundancia presentes en la Base, al almacenar los mismos datos en más de un lugar aumenta la probabilidad de incongruencia ya que al modificar, agregar o eliminar datos se corre el riesgo de no hacerlo en todas las ubicaciones en las que existe, obteniendo una Base inconsistente.

Integra: La integridad de la Base de Datos se refiere a la validez y consistencia de los datos almacenados. Normalmente, la integridad se encuentra definida mediante restricciones o reglas que no pueden ser violadas. Dichas restricciones se pueden aplicar a los datos y a las relaciones existentes entre ellos.

Segura: La seguridad de la Base de Datos es la protección de la misma frente a usuarios no autorizados. Sin unas buenas medidas de seguridad, la integración de datos en los sistemas de Bases de Datos hace que éstos sean más vulnerables que en los sistemas de archivos.

El modelo relacional cumple las siguientes características:

- Independencia física: El modo en que se almacenan los datos no debe influir en su manipulación lógica y, por tanto, los usuarios que acceden a esos datos no han de modificar sus programas por cambios en el almacenamiento físico.
- Independencia lógica: Añadir, eliminar o modificar cualquier elemento de la Base de Datos no debe repercutir en los programas y/o usuarios que están accediendo a subconjuntos parciales de los mismos (vistas).
- Flexibilidad: En el sentido de poder ofrecer a cada usuario los datos de la forma más adecuada a la correspondiente aplicación.
- Uniformidad: Las estructuras lógicas de los datos presentan un aspecto uniforme (tablas), lo que facilita la concepción y manipulación de la Base de Datos por parte de los usuarios.



- Sencillez: Las características anteriores, así como el uso de lenguajes de usuario muy sencillos, producen como resultado que el modelo de datos relacional sea fácil de comprender y de utilizar por parte del usuario final.

Y tiene las siguientes ventajas:

1. Garantiza herramientas para evitar la duplicidad de registros, a través de campos claves o llaves.
2. Garantiza la integridad referencial: así al eliminar un registro elimina todos los registros relacionados dependientes.
3. Favorece la normalización por ser más comprensible y aplicable.
4. Se utiliza a nivel conceptual.

2.1.3 Cardinalidad.

Las Bases de Datos Relacionales se basan en el uso de tablas o relaciones. Gráficamente una tabla se representa por medio de un rectángulo formado por filas y columnas (figura 2.1.3.1), en las columnas se guarda información de las propiedades definidas en la tabla por ejemplo: nombre, apellidos, edad, dirección, etc., a estas propiedades también se les conoce como **atributos**, cada atributo es conocido como **campo**, y en las filas se encuentran representadas las ocurrencias o ejemplares de la instancia o relación que se representa en la tabla, las filas también son conocidas como **tuplas**.

Atributo 1	Atributo 2	Atributo 3	Atributo n	
Valor 1,1	Valor 1,2	Valor 1,3	Valor 1,n	← Tupla 1
Valor 2,1	Valor 2,2	Valor 2,3	Valor 2,n	← Tupla 2
Valor 3,1	Valor 3,2	Valor 3,3	Valor 3,n	← Tupla 3
Valor 4,1	Valor 4,2	Valor 4,3	Valor 4,n	← Tupla 4
Valor m,1	Valor m,2	Valor m,3	Valor m,n	← Tupla m

Figura 2.1.3.1 Representación de una tabla en el modelo relacional.

Las relaciones que se pueden establecer entre los registros o filas de una tabla pueden ser:

Relación 1:1: Se presenta cuando una entidad de tipo A sólo puede estar relacionada con otra de tipo B y no más, y viceversa (figura 2.1.3.2).

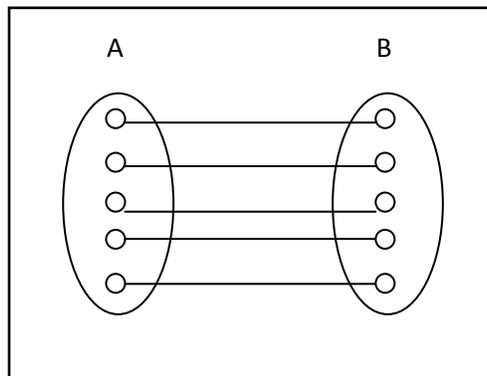


Figura 2.1.3.2 Relación 1:1.

Relación 1:M: En este caso, una entidad de tipo A se puede relacionar con cualquier número de entidades de B, en cambio, una entidad de B sólo puede estar relacionada con una entidad de A, esta relación también es conocida como uno a muchos (figura 2.1.3.3).

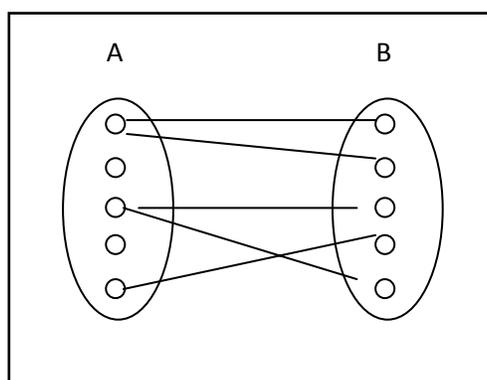


Figura 2.1.3.3 Relación 1:M.

Relación M:M: Establece que cualquier cantidad de entidades del tipo A pueden relacionarse con cualquier cantidad de entidades del tipo B y viceversa, es decir, una relación muchos a muchos (figura 2.1.3.4).

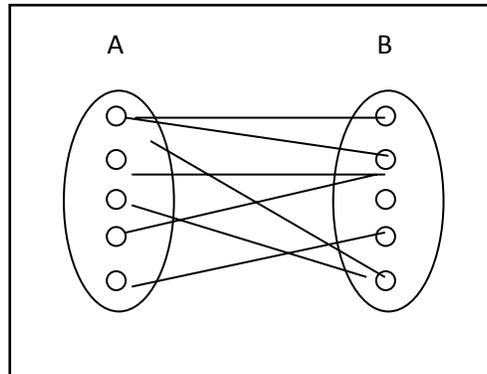


Figura 2.1.3.4 Relación M:M.

2.1.4 Álgebra relacional.

El álgebra relacional, se basa en la teoría de conjuntos y consiste en algunas simples pero poderosas maneras de construir nuevas relaciones a partir de otras ya existentes entre las entidades de una tabla. Si pensamos que las relaciones iniciales son los datos almacenados entonces las nuevas relaciones se pueden ver como respuestas a algunas consultas específicas.

Existen operadores unarios y binarios, para el primer caso tenemos los operadores σ (Selección) y Π (Proyección), el resto de operadores son del segundo tipo. Las operaciones básicas en el álgebra relacional son:

Unión: Se representa como $R \cup S$, es el conjunto de elementos que se encuentran en R, en S o en ambos, en este nuevo conjunto, cada elemento aparece solo una vez.

Intersección: Representado como $R \cap S$, es sólo el conjunto de elementos que aparecen en R y que también aparecen en S.



Diferencia: Se representa como $R - S$ y es el conjunto de elementos que aparecen en A pero no en B, cabe resaltar que el resultado no será el mismo al expresar $R - S$ que en $S - R$.

División: Representado por R / S , es la división de una tupla entre otra en el resultado se debe cumplir que para toda tupla en R, exista su correspondiente en S.

Selección (Select): Se implementa por medio del operador σ , nos permite seleccionar un subconjunto de tuplas dentro de una relación R que cumplan con una condición establecida P y se expresa: $\sigma P(R)$.

Proyección (Project): Permite extraer columnas específicas de una tabla, da como resultado un subconjunto de atributos de la relación original, se expresa: $\Pi_{A1, A2, \dots, An}(R)$, donde A1, A2,, An, son los atributos (columnas) de la relación (tabla) R.

Combinación (Join): Se obtiene al concatenar las tuplas de R y las de S que en los atributos comunes (con el mismo nombre) tienen los mismos valores, estos mismos valores aparecen una sola vez en el resultado, es decir, se eliminan los duplicados.

Producto cartesiano (Product): Se expresa como $R \times S$ y da como resultado una tabla que corresponde a la combinación de las tuplas de R con cada una de las tuplas de S y los atributos serán los de R seguidos por los de S en la nueva tabla.

Diferencia simétrica: Representada como $R \Delta S$, dará como resultado todos los elementos que se encuentran en R y en S, pero eliminando los comunes.

Es posible usar combinaciones de todas las operaciones citadas.



2.1.5 Cálculo relacional.

Es un lenguaje de consulta que describe la respuesta deseada sobre una Base de datos, sin especificar como obtenerla y aquí radica la diferencia con el álgebra relacional, ya que ésta que es de tipo procedural (se le da un procedimiento que debe seguir para resolver el problema) y el cálculo relacional (CR) es de **tipo descriptivo** (se indica cuál es el problema y como resolverlo); pero por ambos métodos se puede conseguir el mismo resultado.

El CR está basado en una rama de la lógica matemática, conocida como Lógica de Predicados o Cálculo de predicados de primer orden; existen dos tipos de CR según el tipo de variables que se manejan y estos son: el cálculo relacional de tuplas (CRT) que usa variables - tupla, que designan a tuplas de relaciones. En el cálculo relacional de dominios (CRD) se utilizan variables – dominio (atributos), que toman valores de los dominios asociados a los atributos de las relaciones.

En el cálculo de predicados (lógica de primer orden), un *predicado* es una función con argumentos que se puede evaluar a verdadero o falso. Cuando los argumentos se sustituyen por valores, la función lleva a una expresión denominada *proposición*, que puede ser verdadera o falsa.

Si un predicado tiene una variable, como en 'x es un empleado de esta empresa', esta variable debe tener un *rango* asociado. Cuando la variable se sustituye por alguno de los valores de su rango, la proposición puede ser cierta; para otros valores puede ser falsa.

En general, todas las operaciones del CRT o del CRD parten de una lógica de primer orden.

- Se puede definir una fórmula en base a combinaciones de fórmulas atómicas.



- Una fórmula atómica es una combinación de variables (tipo tupla o tipo dominio, según corresponda) y atributos o constantes, gracias al uso de operadores como $<$, $>$, $=$, \neq , \leq , \geq .
- Para combinar fórmulas atómicas en caso necesario se hace uso de operadores como NOT (\neg), AND (\wedge), OR (\vee), \rightarrow .
- Una forma de limitar las variables es el uso de cuantificadores tales como \exists (existe), \forall (para todo).

En el **cálculo relacional orientado a tuplas**, lo que interesa es encontrar tuplas para las que se cumple cierto predicado. Una variable tupla es una variable cuyo rango de valores son las tuplas de una relación.

Por ejemplo las expresiones:

$\exists OX (FX.nom_tec = PX.nom_tec \text{ AND } FX.estacion = \text{'zaragoza'})$.

Esta fórmula dice que `existe una falla que tiene el mismo nombre de técnico de la tupla que ahora se encuentra en la variable de Falla, FX, y que está en zaragoza'.

$\forall FX (FX.estacion \neq \text{'zaragoza'})$

Esta fórmula dice que `para todas las tuplas de Falla, la estación no es zaragoza'. Utilizando las reglas de las operaciones lógicas, esta fórmula se puede escribir también del siguiente modo: $\text{NOT } \exists FX (FX.estacion = \text{'zaragoza'})$.

Para en **Cálculo relacional orientado a dominios (CRD)** las variables toman sus valores de dominios, en lugar de tomar valores de tuplas de relaciones. Otra diferencia con el cálculo orientado a tuplas es que en el cálculo orientado a dominios hay un tipo de comparación adicional, a la que se conoce como “*ser miembro de*”, esta condición se expresa de la siguiente forma:



$R(a_1 : v_1, a_2 : V_2)$

Para esta expresión los a_i son los atributos de la relación R y los v_i son las variables o constantes dominio, la condición será verdadera solo si existe una tupla en R que tenga los valores especificados en los atributos declarados, por ejemplo:

FALLA(tecnico: 'joel', estacion: 'candelaria').

Será verdadera si hay algún técnico con el nombre joel y estación candelaria en la tupla FALLA.

En el lenguaje SQL se utilizan algunos aspectos del cálculo relacional orientado a tuplas, pero está basado principalmente en el álgebra relacional.

2.1.6 Reglas de normalización.

La razón para normalizar es asegurar que el modelo conceptual de la base de datos funcionará de manera adecuada. Esto no significa que una estructura no normalizada no funcionará, sino que puede causar algunos problemas cuando los programadores de aplicación traten de modificar la Base de Datos al insertar, actualizar o eliminar datos.

Las ventajas de la normalización son:

- Evita anomalías en inserción, modificación y borrado de datos.
- Mejora la independencia de datos.
- No establece restricciones artificiales en la estructura de los datos.

Uno de los conceptos fundamentales en la normalización es el de dependencia funcional. La dependencia funcional es una noción semántica donde cada dependencia es una clase especial de regla de integridad y representa una relación de uno a muchos.



La normalización se lleva a cabo en una serie de pasos, cada paso corresponde a una forma normal que tiene ciertas propiedades. Conforme se va avanzando en la normalización, las relaciones tienen un formato más estricto y más fuerte y por lo tanto, son menos vulnerables a las anomalías de actualización. Las tres principales formas normales son las siguientes:

- ✓ **Primera Forma Normal (1FN):** Se dice que una relación se encuentra en primera forma normal si cada atributo tiene un valor atómico para cada uno de sus elementos. Para una estructura en forma de tabla existe solo un valor para cada intersección de fila y columna, todos los datos de una columna son del mismo tipo, cada columna tiene un nombre único sin importar el orden de las mismas y por último no debe existir filas repetidas aunque el orden de éstas no es importante.

- ✓ **Segunda Forma Normal (2FN):** Se dice que una relación esta en segunda forma normal si y solo si: la relación cumple con la primera forma normal y además todos los atributos dependen del identificador. Tenemos que validar que cada atributo es dependiente del identificador único. Cada instancia específica del identificador único determina una instancia simple de cada atributo. En caso de que un atributo dependa de una parte del identificador único, entonces está mal ubicado, y este atributo debe de ser movido a otro grupo.

- ✓ **Tercera Forma Normal (3FN):** Debe cumplir con la 2FN y además se eliminan las dependencias transitivas, es decir, ningún atributo considerado no principal depende transitivamente de otro atributo que tampoco esta considerado como llave primaria. En caso de que se presente esta situación, entonces se deben mover los atributos dependientes y el atributo del cual dependen a una nueva entidad, nombrar el atributo del cual dependen, como identificador único y repetirlo en el grupo original para mantener la relación.



Cuando las tablas están en la Tercera Forma Normal se previenen errores de lógica cuando se insertan o borran registros. Cada columna en una tabla está identificada de manera única por la llave primaria, y no debe haber datos repetidos. Esto provee un esquema limpio y elegante, que es fácil de trabajar y expandir.

Las reglas de normalización están encaminadas a eliminar redundancias e inconsistencias de dependencia en el diseño de las tablas.

Una **llave primaria** es aquel atributo que consideramos clave para la identificación de los demás atributos que describen a la entidad.

Puede haber más de un atributo que pueda ser identificado como llave primaria en este caso se selecciona el que consideremos más importante, los demás atributos son denominados **llaves secundarias**.

2.1.7 Metodologías de diseño.

MERISE.

En esta metodología de origen francés se hace una descripción del sistema en tres niveles:

- **Conceptual:** En este nivel se tratan básicamente los datos y la manera en que estos serán manipulados. Se describe como los datos serán reflejados en la información existente en su entorno y las relaciones entre ellas.
- **Lógico y organizativo:** Aquí se desglosan las operaciones descritas en el nivel conceptual en procedimientos funcionales, es decir, en tareas realizadas sucesivamente en un puesto de trabajo. En este nivel se definen las tareas que hay que realizar.

- **Físico y operativo:** En este nivel se indica cómo hay que realizar las actividades asignadas. Se detallan las indicaciones para todo tipo de procedimientos, así como las normas para realizarlos correctamente.

Las fases de la metodología MERISE son:

1. Estudio preliminar.
2. Estudio detallado.
3. Implementación.
4. Realización y puesta en marcha.

Estas etapas se encuentran representadas en la figura 2.1.7.1.

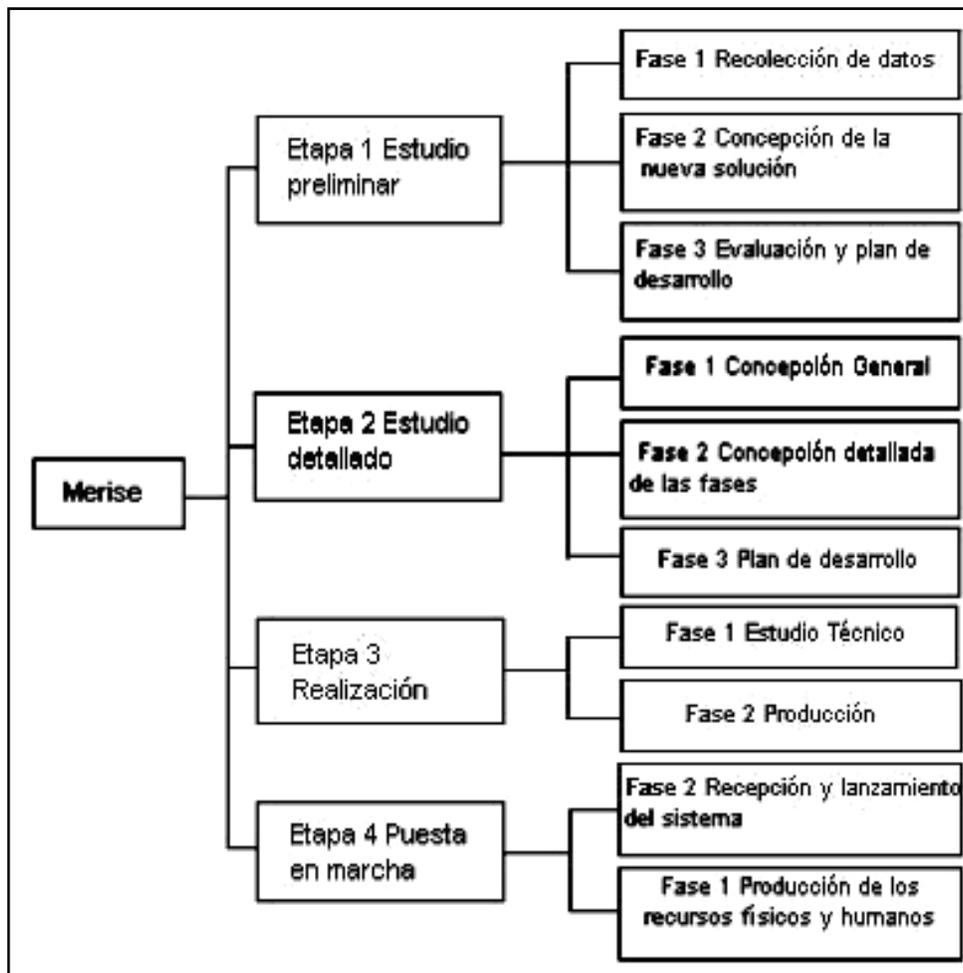


Figura 2.1.7.1 Diagrama representativo de la metodología MERISE.



SSADM.

La metodología SSADM (Structured Systems Analysis and Design Method / Método de Diseño y análisis de Sistema Estructurado), consiste en una estructuración de los pasos a seguir en el desarrollo de un sistema de información en las fases iniciales y en la descripción de las técnicas y formalismos sobre las que se basan los trabajos a realizar en cada fase.

En esta metodología se usan esquemas en los que se puede ver la secuencia de los pasos y las técnicas utilizadas en cada uno de ellos dentro de cada fase.

Los esquemas que se implementan son:

DFD Es la forma de representación de los flujos de información a nivel interno y con el exterior del sistema, es decir, con sistemas externos.

Estructura Lógica de Datos (LDS): Esto se hace mediante la representación de las entidades del sistema y las relaciones entre ellas. Es decir utiliza el formato de modelo E/R.

Historia en la Vida de la Entidad (ELH): Representa la descripción de cómo las entidades descritas son afectadas por diferentes sucesos que ocurren en el sistema. La metodología SSADM plantea las etapas, fases y pasos a seguir en el desarrollo de un sistema como se representa en la figura 2.1.7.2.

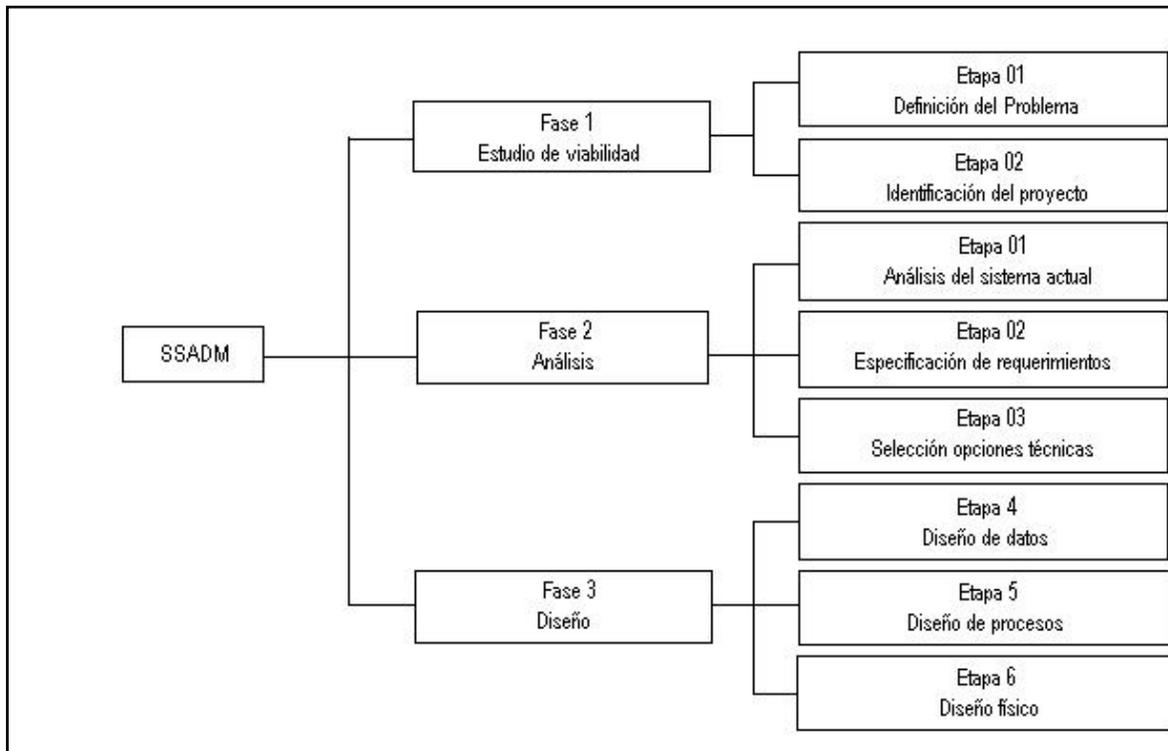


Figura 2.1.7.2 Representación gráfica de la metodología SSADM.

YOURDON / DE MARCO.

Se considera como una metodología moderna en el desarrollo de sistemas, las actividades a realizar en la metodología son:

- **Estudio preliminar o de factibilidad del negocio:** Es la representación al nivel de diagramas de flujo de la identificación de puntos críticos y de procesos a automatizar.
- **Análisis estructurado:** En esta etapa se hace la representación del sistema a desarrollar, mediante los diagramas conocidos como DFD, DTE, Diagramas de Estructura, Etc.

Diagrama de flujo de datos (DFD): En él se representan los procesos o funciones que lleva a cabo el sistema en diferentes niveles y los datos que fluyen entre las mismas funciones.



El diagrama de transición de estados (DTE): Enfatiza el comportamiento del sistema en función del tiempo (en tiempo real).

Hasta hace un tiempo, los modelos del comportamiento dependiente del tiempo del sistema importaban solo para una categoría especial de sistemas conocidos como sistemas de tiempo real, por ejemplo sistemas de conmutación telefónica.

Para sistemas enfocados a los negocios no se veían demasiado importante, sin embargo en sistemas grandes y complejos enfocados a negocios que si tienen aspectos de comportamiento de tiempo real. Aunque no se apliquen en todos los sistemas es conveniente estar familiarizado con herramientas de modelado para el comportamiento dependiente del tiempo.

- **Diseño:** En esta etapa se transforma lo obtenido a través del modelo E/R en el diseño de la base de datos, es decir, se pasa del nivel conceptual a un nivel de representación lógica de los datos.
- **Implantación:** Esta actividad incluye la programación e integración de los módulos del sistema.
- **Pruebas de aceptación:** En esta fase se hacen pruebas de integración y de funcionamiento conjunto de los programas.
- **Garantía de calidad:** Esta etapa es la prueba final de calidad del sistema ya que se debe considerar llevar a cabo actividades de control durante todas las etapas anteriores para asegurar que todas y cada una de ellas cuenten con niveles óptimos de calidad.
- **Elaboración de manuales de usuario:** En esta etapa se documentan todos los procesos de la aplicación. Aquí se crea el documento que detalla el modo de operar el sistema, para trabajar con él adecuadamente y explotando sus funciones al máximo.
- **Conversión de datos:** En caso de que exista un sistema anterior, se desarrollan interfaces para la migración de datos de un sistema a otro.

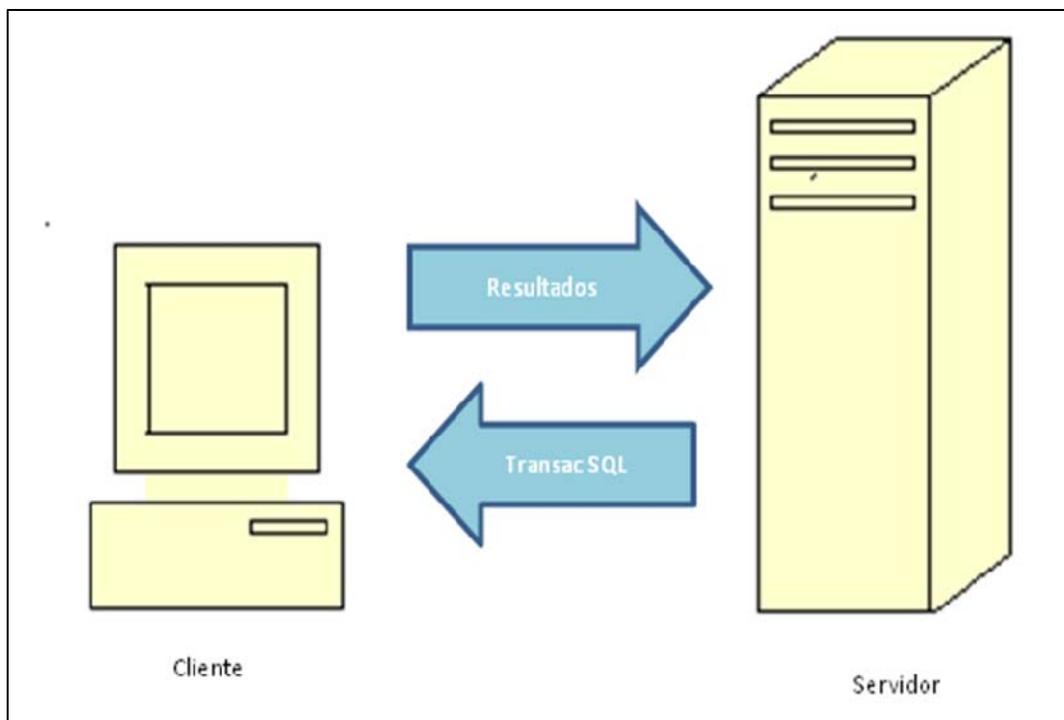


- **Instalación:** Comprende la puesta en producción del sistema y la captación de los usuarios.
- **Mantenimiento:** Comprende la proyección a largo plazo del mantenimiento que se le deberá dar al sistema.

2.2 Características, ventajas y desventajas de SQL Server 2005®.

2.2.1 Introducción.

SQL (Structured Query Language, Lenguaje de Consulta Estructurado) es un sistema administrador para Bases de Datos relacionales basadas en la arquitectura Cliente – Servidor, que usa Transact-SQL para mandar peticiones entre un cliente y el SQL Server. En la figura 2.2.1.1 se muestra el funcionamiento lógico que utiliza SQL Server.



2.2.1.1. Funcionamiento Lógico de SQL Server.

SQL es utilizado por los diferentes motores de bases de datos para realizar determinadas operaciones sobre el contenido o sobre la estructura, por lo que se



dice que es un Sistema Administrador para Base de Datos Relacionales (RDBMS) ya que mediante su gestión permite:

- 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.

Para la creación de nuestra Base de Datos utilizaremos SQL debido a que tiene grandes ventajas en comparación con otros potentes Sistemas gestores de Base de Datos como son: Oracle, Sybase ASE, PostgreSQL, Interbase, Firebird o MySQL , ya que aunque no es un Sistema multiplataforma como los antes mencionados y solo esta disponible en Sistemas Operativos de Microsoft , el software de nuestro Sistema a desarrollar esta basado en Microsoft, lo cual reduce los costos considerablemente ya que no será necesario utilizar otra tecnología y se pueden explotar los recursos actuales de la empresa.

2.2.2 Características de SQL.

El SQL es un lenguaje declarativo de alto nivel, que gracias a su fuerte base teórica y su orientación al manejo de conjuntos de registros, y no a registros individuales, permite una alta productividad en codificación. De esta forma una sola sentencia puede equivaler a uno o más programas que utilizasen un lenguaje de bajo nivel orientado a registros.

Entre las principales características de SQL se encuentran las siguientes:

- SQL Server posee una gran variedad de herramientas administrativas y de desarrollo que permite mejorar la capacidad de instalar, distribuir, administrar y utilizar SQL Server.
 - Administrativas (Administrador Corporativo).



– Desarrollo (Analizador de consultas).

- Puede utilizarse el mismo motor de base de datos a través de plataformas que van desde equipos portátiles que ejecutan Microsoft Windows 95, 98 ó XP hasta grandes servidores con varios procesadores que ejecutan Microsoft Windows NT, Enterprise Edition.
- Almacenamiento de Datos.
- SQL Server incluye herramientas para extraer y analizar datos resumidos para el proceso analítico en línea (OLAP, Online Analytical Processing) y para el procesamiento de transacciones en línea (OLTP, Online Transaction Processing).

Sistemas OLAP.

La tecnología OLAP permite un uso más eficaz de los almacenes de datos para el análisis en línea, lo que proporciona respuestas rápidas a consultas analíticas complejas e iterativas. Los modelos de datos multidimensionales de OLAP y las técnicas de agregados de datos organizan y resumen grandes cantidades de datos para que puedan ser evaluados con rapidez mediante el análisis en línea.

Sistemas OLTP.

Los sistemas OLTP están diseñados y ajustados para procesar cientos o miles de transacciones que se introducen al mismo tiempo. Los datos son generalmente organizados en tablas relacionales para evitar redundancia e incrementar la velocidad de las actualizaciones.

- **SQL** incluye también un potente entorno gráfico de administración, que permite el uso de comandos DDL y DML gráficamente.

Comandos DDL.

Los comandos DDL son las siglas de Data Definition Language (Lenguaje de Definición de Datos), y se corresponde con el conjunto de órdenes que permiten definir las estructuras que van a contener los datos en un repositorio. Como



concepto, es ajeno a un tipo de repositorio concreto, ya sea una base de datos relacional, un OLAP o XML.

La tabla 2.2.2.1 muestra los comandos DDL básicos:

COMANDO	DESCRIPCIÓN
CREATE	Utilizado para crear nuevas tablas, stored procedures e índices.
DROP	Empleado para eliminar tablas, stored procedures e índices.
ALTER	Utilizado para modificar las tablas agregando campos o cambiando la definición de los campos.

2.2.2.1 Comandos DDL básicos.

Comandos DML.

Los comandos DML son las siglas de Data Manipulation Language (Lenguaje Manipulador de Datos) y se refiere a los comandos que permiten a un usuario manipular los datos en un repositorio, es decir, añadir, consultar, borrar o actualizar.

La tabla 2.2.2.2 muestra los comandos DML básicos:



COMANDO	DESCRIPCIÓN
SELECT	Utilizado para consultar registros de la base de datos que satisfagan un criterio determinado.
INSERT	Utilizado para cargar lotes de datos en la base de datos en una única operación.
DELETE	Utilizado para modificar los valores de los campos y registros especificados.
UPDATE	Utilizado para eliminar registros de una tabla de una base de datos.

2.2.2.2 Comandos DML básicos.

- **SQL** Server 2005 se integra con el correo electrónico, internet y Windows, permitiendo una comunicación local.

Dentro del SQL podemos encontrar dos variantes de éste, el SQL estático y el dinámico. Podríamos describir al SQL estático como aquella parte del mismo que se ocupa de consultar, modificar, etc., una base de datos en la cual se puede decir que la forma de actuar es siempre la misma; las tablas y columnas referenciadas son, por tanto, conocidas de antemano por el programador, lo que le lleva a crear una serie de pasos comunes a seguir. Esto supone que si en algún momento fuese el propio programa creado el que tuviese que decidir, en tiempo de ejecución, las sentencias a usar o las tablas a referenciar, este tipo de SQL nos sería completamente inútil.

Por ello, se crea lo que se conoce como SQL dinámico, capaz de no codificar de forma fija una sentencia de SQL inmerso en un código fuente. Por su parte, hace que sea el programa el que construya dicha(s) sentencia(s), en una de sus áreas



de datos en tiempo de ejecución, y luego traspase el texto de las(s) misma(s) a la base de datos para que se ejecute en tiempo real.

2.2.3 Ventajas de SQL Server 2005.

- Facilidad de instalación, distribución y utilización.
- Soporte de Transacciones.
- Escalabilidad, estabilidad y seguridad.
- Soporta procedimientos almacenados.
- Permite trabajar en modo cliente – servidor, donde la información y datos se alojan en el servidor y las terminales o clientes de la red sólo acceden a la información.
- Permite administrar información de otros servidores de datos.
- Los datos en SQL Server 2005 pueden ser automáticamente exportados a otros sistemas independientes (Oracle, Sybase, Access, etc.).
- Bajo costo en comparación con otros productos.
- Integración con Internet.
- Soporta autoconfiguración.

2.2.4 Desventajas de SQL Server 2005.

- Costo de las licencias comparadas con otros competidores.
- SQL Server es dependiente de la plataforma Windows.
- No permite elaborar formas de edición de datos, reportes menús de operaciones, etc. Se debe apoyar de otras herramientas.



2.3 Características, ventajas y desventajas de Microsoft Visual Basic®.

2.3.1 Introducción.

Visual Basic es un ambiente gráfico de desarrollo de aplicaciones para el sistema operativo Microsoft Windows®. Las aplicaciones creadas con Visual Basic® están basadas en la programación orientada a objetos y son manipuladas por medio de eventos. Visual Basic® se deriva del Lenguaje Basic, el cual se caracteriza por estar basado en programación estructurada, pero con una secuencia de ejecución controlada por eventos.

En las aplicaciones tradicionales o procedurales, es la aplicación misma quien controla cuáles segmentos de código son ejecutados y con qué secuencia, es decir, la ejecución de una aplicación se inicia en la primera línea de código, y sigue una ruta que ha sido predefinida a través de la aplicación, llamando procedimientos según sea necesario.

En las aplicaciones que son manejadas por eventos, la ejecución de código no sigue una ruta predefinida. La ejecución de las diferentes secciones de código que conforman a la aplicación se dan en respuesta a eventos. Los eventos se desencadenan como resultado de las acciones del usuario al interactuar con la aplicación, por mensajes del sistema o de otras aplicaciones. La secuencia en que los eventos son provocados determina el modo y secuencia en que el código se ejecuta y es por esto que la ruta que sigue el código de la aplicación puede o no, variar cada vez que se ejecuta el programa.

2.3.2 Características.

Es fundamental en la programación manejada por eventos escribir código que responda a posibles ocurrencias que puedan presentarse en la aplicación, es decir, tener la visión adecuada para prevenir todos y cada uno de los posibles caminos que la ejecución podrá tomar en respuesta a los diversos eventos que se puedan dar.



Los bloques básicos para la construcción de una aplicación por medio de Visual Basic® son: los formularios (ventanas), los menús y los controles, también son denominados objetos, es decir, representan los bloques básicos que se pueden usar al desarrollar una aplicación de Visual Basic®.

Cada objeto posee un conjunto de características y un comportamiento definido por propiedades, métodos y eventos que lo distinguen de otros tipos de objetos, es decir, el objeto formulario está diseñado para cumplir determinada función en una aplicación, y no es lo mismo que un objeto menú o un control.

Al conjunto de datos que describen las características de un objeto se le conoce como sus propiedades. Para un formulario tenemos por ejemplo, propiedades que permiten cambiar su aspecto, tales como: Font (tipo de fuente), BackColor (color de fondo) o Height (altura); algunas propiedades determinan aspectos diferentes del objeto, por ejemplo, la propiedad MaxButton establece si en el formulario estará presente el botón *Maximizar*, es decir, si el formulario se puede maximizar o no.

Los métodos son un conjunto de procedimientos que permiten que un objeto ejecute una acción o tarea sobre sí mismo. Por ejemplo, para un formulario tenemos el método Hide que hará que el formulario se oculte; o el método Show, que hará que este se vuelva a mostrar.

Un evento es una acción que es reconocida por el objeto. Un evento ocurre (se dispara) como resultado de la interacción del usuario con el objeto, debido a la ejecución de código (sentencias), o como resultado de la interacción de otro objeto con el objeto poseedor del evento.

Para un formulario tenemos por ejemplo, el evento Load que se dispara cuando se carga el formulario, o el evento Click para un botón de comando que se dispara cuando se hace clic sobre él.



Toda aplicación necesita una interfaz de usuario, la parte visual a través de la cual el usuario interactúa con esta. Los bloques básicos de construcción de una interfaz de usuario son los formularios y los controles. Visual Basic utiliza técnicas de programación visual para diseñar las aplicaciones.

Visual Basic 6.0® es una herramienta de desarrollo diseñada para crear aplicaciones con interfaz gráfica y utiliza fundamentalmente dos ventanas, una que permite realizar los diseños gráficos y un lenguaje de alto nivel. La palabra “Visual” hace referencia al método que utiliza para crear la interfaz gráfica de usuario. En lugar de escribir numerosas líneas de código para implementar una interfaz, se utiliza el Mouse para arrastrar y colocar los objetos prefabricados al lugar deseado dentro de un formulario.

La palabra “Basic” hace referencia al lenguaje BASIC “Beginners All-Purpose Symbolic Instruction Code”. Visual Basic ha evolucionado a partir del lenguaje Basic original y ahora contiene varias instrucciones, funciones y palabras clave, muchas de las cuales están directamente relacionadas con la interfaz gráfica de Windows.

A continuación se muestra el aspecto típico de la pantalla inicial al cargar Visual Basic 6.0® (figura 2.3.2.1). En esta ventana se seleccionará el tipo de proyecto que se desea desarrollar, generalmente se seleccionará la opción “EXE estándar”.

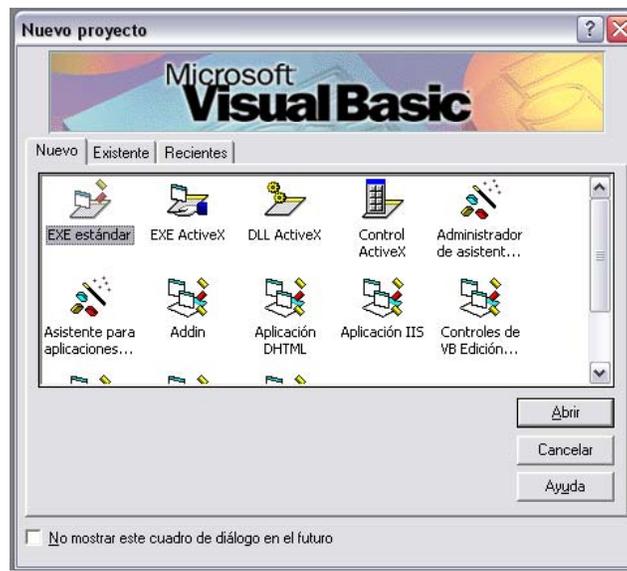


Figura 2.3.2.1 Ventana inicial de Visual Basic ®.

Una vez seleccionado el tipo proyecto a desarrollar, se muestra una nueva con el entorno de trabajo de Visual Basic ® (figura 2.3.2.2), dentro del cual se puede hacer uso de las siguientes ventanas:

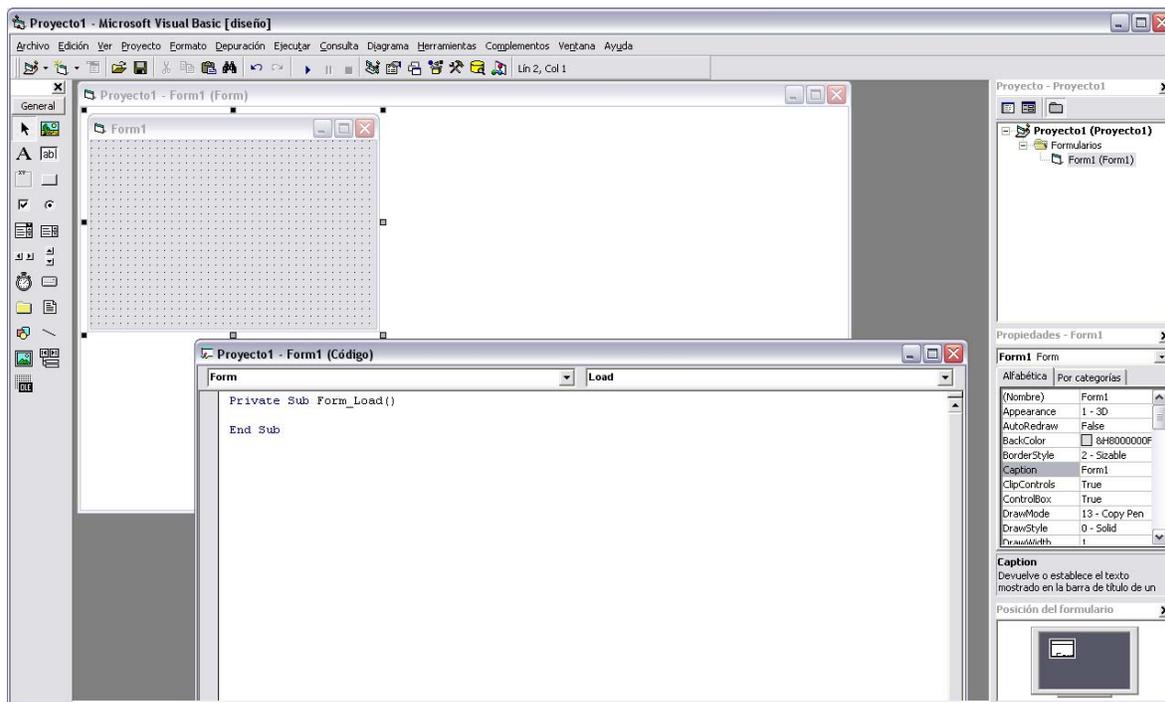


Figura 2.3.2.2 Entorno de trabajo de Visual Basic®



- **Formulario:** Es la parte principal de una aplicación basada en gráficos. Es lo que el usuario ve y con lo cual interactúa para llevar a cabo alguna tarea. También es el lugar en donde se comienza a construir una aplicación, es decir, donde se colocan los controles para construir la interface.
- **Proyecto:** Es una lista de archivos que se usan para llevar el control de los formularios y módulos que conforman una aplicación, en ella se encuentran archivos con extensión FRM asociados a los formularios, VBP ligados al proyecto, .BAS que contienen código, .CLS con definiciones de clases, etc.
- **Propiedades:** Muestra las características del control seleccionado en la formulario.
- **Caja de Herramientas:** Es el lugar en donde se encuentran todos los posibles controles que se pueden dibujar en una aplicación. Existen dos maneras de colocar controles en un formulario, la primera es por medio de un doble clic en el control que se desee dibujar, con esto automáticamente el control se colocará al centro del formulario y con un tamaño predefinido. Otro método es por medio de un clic, y arrastrando el apuntador del Mouse sobre el formulario hasta que el control tenga el tamaño y la posición deseada.
- **Barra de Herramientas:** La barra de herramientas permite un acceso rápido a las funciones y comandos comunes, estas funciones y comandos también están disponibles en los menús o pueden ser accedidos por medio de teclas especiales.
- **Ventana de código:** En esta ventana se muestra una expansión del código que implementa una aplicación. Al principio las ventanas de código sólo contienen una plantilla para los procedimientos, en las cuales se agregará el código necesario para que una aplicación funcione. Para poder ver la ventana de código se debe dar un doble clic sobre el objeto o control para el que se va a desarrollar código.
- **Posición del formulario:** Muestra a escala la ubicación en donde se encontraría el formulario al momento de ejecutar la aplicación.



- **Timer:** Por medio de este control el programador puede definir eventos que se ejecutarán cada determinado tiempo.
- **Cuadros de Imagen:** Son utilizadas para desplegar imágenes en el formulario, éstas pueden ser iconos, fondos, botones, etc.

Algunas de las características más sobresalientes de Visual Basic ® son:

- Biblioteca de clases que brinda soporte a los objetos Windows tales como ventanas, cajas de diálogo, controles, etc.
- Controles que permiten utilizar las cajas de diálogo que comúnmente se utilizan en las aplicaciones tipo Windows para abrir, guardar, imprimir, color y fuentes.
- Un entorno de desarrollo completo e integrado (editor de código, intérprete, depurador, examinador de objetos, explorador de proyectos, compilador, etc.). Visual Basic® es un intérprete, lo que permite crear y depurar las aplicaciones, a partir de la versión 5.0, se incluyó un compilador que nos permite generar archivos .exe que favorecen la ejecución sin proporcionar el código fuente. Puede crear archivos .ocx y .dll lo que permite manipular controles ActiveX.
- El editor de código completa cada una de las instrucciones al tiempo de estar programando, visualizando la sintaxis correspondiente de las mismas en tiempo real.
- Cuenta con asistentes para el desarrollo de aplicaciones, empaquetado y distribución, páginas de propiedades, objetos de datos, generador de clases, diseñador de complementos y para migración de documentos ActiveX.
- Visualización y manipulación de datos de otras aplicaciones Windows utilizando controles OLE (Object linking and embedding).
- Una interfaz para múltiples documentos MDI, que permite crear una aplicación con una ventana principal y múltiples ventanas de documento que se abren dentro de ésta.



- Permite, durante una sesión de depuración realizar modificaciones en el código de la aplicación sin tener que salir de dicha sesión.
- Creación y utilización de bibliotecas dinámicas DLL (Dynamic Link Libraries/Biblioteca de vínculos dinámicos).
- Facilita la programación de aplicaciones para Internet; mediante la tecnología de componentes activos ActiveX.
- Soporte para el estándar COM (Componente Object Model/Modelo de Componentes de Objetos).

2.3.3 Ventajas.

La ventaja principal que brinda este lenguaje de programación es la facilidad para programar aplicaciones de cierta complejidad para Windows, así como los requerimientos mínimos que para su instalación requiere, y de los cuales ya se hablo con anterioridad.

Algunas ventajas adicionales son:

- Es un lenguaje de Desarrollo Rápido de Aplicaciones (RAD).
- Es un lenguaje fácil y rápido de aprender.
- Los formularios en el estilo de Windows® son fáciles de diseñar e implementar.
- La plataforma de los diversos Sistemas Windows® se usa con gran facilidad.
- El código que se desarrolla con Visual Basic®, es de fácil migración a otros lenguajes.

Para crear una nueva aplicación con Visual Basic®, se deben seguir estos tres pasos fundamentales:

- Crear una interfaz gráfica con la que el usuario interactúe.
- Definir las propiedades de los objetos.
- Definir el código asociado a cada uno de los objetos.



2.3.4 Desventajas.

A continuación un listado de las principales desventajas de Visual Basic®:

- La necesidad de archivos adicionales además del ejecutable, lo cual implica una cierta lentitud en el momento de la ejecución en comparación con otros lenguajes pero esto hoy en día es cada vez menos determinante debido a la velocidad y capacidad del hardware de última generación.
- Sólo corre bajo sistema operativo Windows®, aunque se han desarrollado emuladores e interpretes para poder correrlo en otros S.O.
- Permite la programación sin declarar variables, este inconveniente puede ser resuelto al escribir “Option Explicit” en el encabezado de cada formulario, con esto se hace necesaria la declaración de variables y se obtiene un código más puro.
- Solo se permite el uso de funciones definidas en librerías dinámicas (DLL).
- Existen funciones sin documentación (pero esto ocurre en muchos otros lenguajes).
- El lenguaje esta basado en objetos, pero no orientado a ellos.
- Manejo inadecuado en los apuntadores de memoria.
- No existen operadores de desplazamiento de bits nativos.
- No permite el uso de memoria dinámica.
- Puede no avisar de errores como desbordamientos.

2.4 Características, ventajas y desventajas de Crystal Reports.

2.4.1 Características de Crystal Reports.

Crystal Reports es una herramienta especializada en la generación de informes para el entorno Windows, que mediante un conjunto de diseñadores y asistentes nos permite la creación de los listados para nuestras aplicaciones de un modo muy sencillo, y con un resultado y apariencia profesionales.

Se trata de un generador de informes que ha acompañado a Visual Basic desde hace varias versiones.



Crystal Reports es un líder comprobado en el diseño de reportes que cumplan con los desafíos que día a día enfrentan los analistas y los desarrolladores.

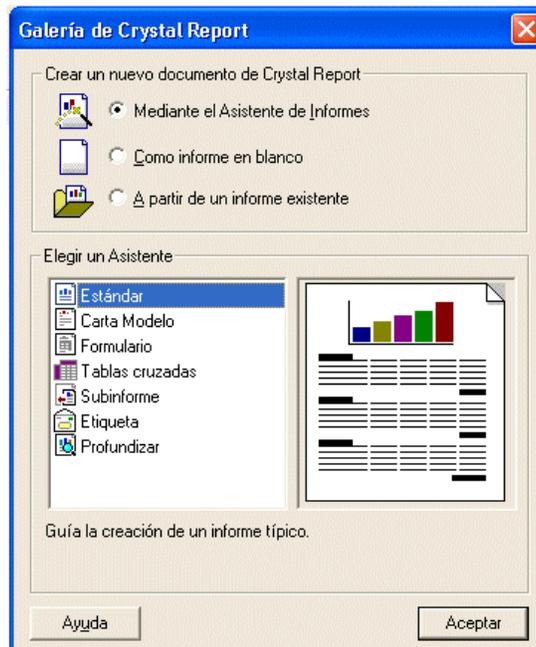
Crystal Reports nos permite:

- Transformar rápidamente cualquier fuente de datos en contenido interactivo.
- Integrar estrechamente capacidades de diseño, modificación y visualización en aplicaciones .NET, Java o COM.
- Permite a los usuarios finales acceder e interactuar con los reportes a través de portales Web, dispositivos móviles y documentos de Microsoft Office.

Crystal Reports mediante su ventana de Galería de Crystal Reports (Vea figura 2.4.1.1) nos permite crear reportes mediante un asistente de Informes y de forma manual.

La creación de un informe a través del asistente, es un proceso que simplifica en gran medida el trabajo del programador, evitando que este se ocupe entre otros aspectos del diseño, de la organización y ubicación de los campos en el informe. Sin embargo, nos encontraremos con situaciones en las que la disposición automática de los campos generada por el asistente, y otros aspectos de su configuración, no se adaptarán a nuestras necesidades.

En este tipo de casos deberemos optar por crear nosotros mismos el informe partiendo de la plantilla del diseñador vacía, y establecer la fuente de datos, campos, y demás objetos del listado.



2.4.1.1 Selección del modo de creación y asistente del informe El diseñador de informes.

Esta ventana nos permite el diseño visual del informe de una manera muy fácil y cómoda, con el mismo estilo de trabajo que el usado con el diseñador de formularios Windows.

Cuando utilizamos el Asistente de Informes, el asistente crea un conjunto de controles dentro del diseñador en forma de campos, que mostrarán los datos cuando sea ejecutado el informe.

El contenido actual del informe también puede ser modificado, ya que lo que hemos obtenido con el asistente ha sido un diseño basado en una guía predefinida por Crystal Reports, y que no siempre coincidirá con el diseño que teníamos previsto realizar para nuestro listado, por lo que podemos añadir, modificar, quitar campos y otros controles del informe.



Un informe consta de una serie de secciones, dentro de las cuales se organiza la información. Las secciones básicas que encontraremos habitualmente en todo informe serán las siguientes:

- **Encabezado del informe.** Esta sección se imprime una vez al comienzo de la ejecución del informe, siendo adecuada para insertar en ella información general como título, autor, imágenes, gráficos, etc.
- **Encabezado de página.** Esta sección se imprime una vez por cada página del informe, y en ella podemos situar los títulos de las columnas de los campos de la tabla a imprimir, la fecha del listado, etc.
- **Detalles.** Es la sección más importante, y en ella se sitúan los campos de la tabla que representan los datos del informe, y que se imprimirán tantas veces como registros sea necesario mostrar.
- **Pie del informe.** Se imprime una vez al final del informe, siendo una sección propicia para insertar un total general que muestre el resultado de la suma de columnas numéricas.
- **Pie de página.** Se imprime una vez por página, justamente después de las líneas de detalle. Se trata de una sección muy adecuada para incluir números de página, líneas separadoras, etc.

El control CrystalReportViewer.

El control CrystalReportViewer, como indica su nombre, se trata de un visualizador de informes que nos permitirá mostrar el listado en un formulario, para su comprobación por parte del usuario.

El control CrystalReportViewer incorpora en su barra de herramientas una serie de funcionalidades estándar, que evitan al programador tener que implementarlas en su aplicación, entre ellas se encuentran la posibilidad de navegar entre las páginas del informe, exportarlo a diversos formatos de documento (PDF, Word, Excel, etc.), buscar texto, etc.



La clase del informe.

Al igual que ocurre con los formularios Windows, los informes de Crystal Reports disponen de un diseñador, para crearlos visualmente, y de una clase, que hereda de la clase ReportClass, y proporciona la funcionalidad necesaria.

El hecho de que un informe sea internamente una clase, nos proporciona una gran flexibilidad, ya que nos permite crear un objeto a partir de la misma, y manipular el informe a través de sus propiedades y métodos.

Clase ReportDocument.

La clase ReportDocument representa un objeto informe, y dispone de los miembros para efectuar todas las operaciones de manipulación, de igual forma que si utilizáramos directamente un objeto instanciado de la clase del informe. Una vez soltado el componente ReportDocument sobre una plantilla de formulario, el motor de informes de Crystal Reports busca en el proyecto la existencia de informes diseñados, y nos muestra un cuadro de diálogo en el que se solicita asociar el ReportDocument que estamos agregando al formulario con un informe existente, eligiéndolo en una lista desplegable.

Tipos de Informes.

En función de cómo los informes sean manipulados desde un proyecto, Crystal Reports nos permite clasificarlos en las siguientes categorías:

- **Informe con establecimiento inflexible de tipos (Strong typed).** Todo informe que constituya parte de un proyecto, es decir, que aparezca al igual que lo hace en el Explorador de soluciones un formulario, clase, etc., se dice que es un informe con *establecimiento inflexible de tipos* o *strong typed*. Este tipo de informe, es manejable dentro del proyecto como una clase, pudiendo crear instancias de la misma.



- **Informe sin tipo (Un-Typed).** Todo informe que no constituya parte de un proyecto, y que sea cargado de forma externa en tiempo de ejecución por la aplicación se dice que es un *informe sin tipo* o *un-typed*.
- **Informe sin tipo cargado desde el control CrystalReportViewer.** Esta es la técnica más sencilla, y consiste en asignar a la propiedad ReportSource del control CrystalReportViewer incluido en un formulario, una cadena con la ruta de un archivo .RPT que tenga un informe diseñado con Crystal Reports.
- **Informe sin tipo cargado desde el componente ReportDocument.** El usuario pensará con toda probabilidad que esta técnica es más complicada para, a fin de cuentas, obtener el mismo resultado. La ventaja radica en que, a través del componente ReportDocument, obtenemos una mayor flexibilidad a la hora de manipular el contenido del informe por código.

2.4.2 Ventajas de Crystal Reports.

- Creación de reportes de alta productividad.
- El más completo acceso a datos: Crystal Reports provee más opciones de conectividad a datos que cualquier otra herramienta. Incluye más de 30 drivers para acceso a bases de datos relacionales, fuentes de datos XML y cubos OLAP (Incluyendo sistemas ERP, CRM, Oracle, IBM DB2 y Microsoft SQL Server). También puede acceder a datos personalizados a través de JavaBeans y objetos COM (ADO record sets) para una conectividad más flexible.
- Diseño integral y opciones de formato: La tecnología flexible de diseño de Crystal Reports provee control completo sobre el acceso y la presentación de los datos en los reportes. Se puede elegir entre más de 100 opciones de formato, incluyendo parámetros, mapas, tablas cruzadas, gráficos e hipervínculos, para incrementar el impacto de sus reportes. También se incluyen más de 160 formulas, funciones y operadores para un control completo de la presentación de los datos.



- Productividad: El Repositorio Crystal permite almacenar elementos clave de los reportes de su organización, tales como objetos de texto, imágenes, sentencias SQL y funciones personalizadas. Gracias a este repositorio central se puede reutilizar estos objetos en múltiples reportes. Este almacén centralizado de objetos le permite minimizar los esfuerzos de mantenimiento de sus reportes y al mismo tiempo ser más productivo en el diseño de reportes nuevos.

De la misma manera, se pueden crear plantillas corporativas para ser aplicadas a todos los reportes de una organización y así evitar dar formato una y otra vez a cada uno de los nuevos reportes que se crean.

- La interacción y acceso a los datos que los usuarios demandan: Crystal Reports habilita la visualización e interacción con los reportes a través de una amplia variedad de dispositivos y entornos. Se utiliza los Report Parts (fracciones de reporte) o el soporte para Smart Tags en Office XP, para proveer acceso, navegación e interacción con los reportes vía portales web, documentos de Microsoft Office o dispositivos inalámbricos.
- Los visores Web avanzados habilitan a los usuarios finales para realizar búsquedas dentro de los datos de un reporte y exportarlas posteriormente a Microsoft Excel, Word y páginas HTML con el vínculo dinámico al reporte original. Adicionalmente, el reporte completo puede ser exportado a una variedad de formatos incluyendo XML, PDF, HTML y Microsoft Excel.
- La flexibilidad y control: Con flexibles SDKs (Software Development Kits) para aplicaciones .NET, Java y COM y controles para que incluso el usuario final pueda hacer cambios a los reportes, Crystal Reports habilita la estrecha integración tanto en aplicaciones Cliente/Servidor como en aplicaciones Web.
- Flexibilidad en Integración e Interacción.



2.4.3 Desventajas de Crystal Reports.

- Costo de licencia
- Únicamente entorno Windows

2.5 Características, ventajas y desventajas de la Arquitectura Cliente - Servidor.

2.5.1 Introducción

Una arquitectura es una serie de elementos funcionales que aprovechando diferentes estándares, convenciones, reglas y procesos permite integrar una amplia gama de productos y servicios informáticos, de manera que pueden ser utilizados eficazmente dentro de la organización, para seleccionar el modelo de una arquitectura, hay que partir del contexto tecnológico y organizativo del momento ya que la arquitectura Cliente/Servidor requiere una determinada especialización de cada uno de los diferentes componentes que la integran.

Cliente es el que inicia un requerimiento de servicio, donde, el requerimiento inicial puede convertirse en múltiples requerimientos de trabajo y sin importar la ubicación de los datos o de las aplicaciones la información obtenida es totalmente transparente en base a la solicitud realizada al servidor, a este proceso se le conoce con el término de Front-End.

Las funciones que lleva a cabo el proceso cliente son:

- Administrar la interfaz del usuario.
- Interactuar con el usuario.
- Procesar la lógica de la aplicación y hacer validaciones locales.
- Mostrar los resultados del servidor.
- Formatear resultados.

Servidor es cualquier recurso de cómputo dedicado a responder a los requerimientos del cliente, los servidores pueden estar conectados a los clientes a



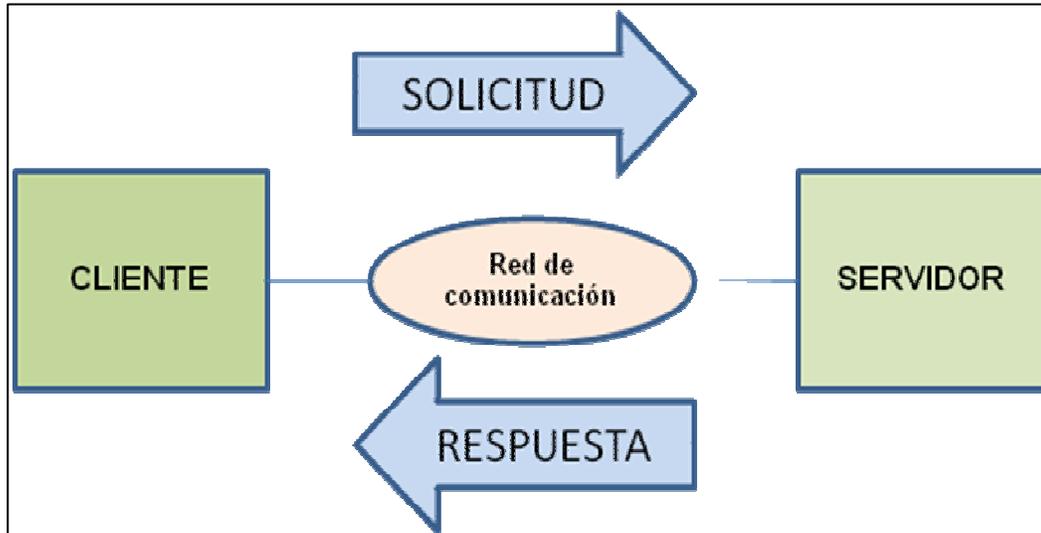
través de redes para proveer de múltiples servicios a los clientes tales como impresión, acceso a bases de datos, fax, procesamiento de imágenes, reportes, etc. A este proceso se le conoce como Back-End.

Las funciones que lleva a cabo el proceso del servidor son:

- Aceptar los requerimientos de bases de datos que hacen los clientes.
- Procesar requerimientos de bases de datos.
- Formatear datos para transmitirlos a los clientes.
- Procesar la lógica de la aplicación y realizar validaciones a nivel de bases de datos Middleware.

La arquitectura cliente/servidor se divide en dos partes claramente diferenciadas, la primera es la parte del servidor y la segunda la de un conjunto de clientes, normalmente el servidor es una máquina bastante potente que actúa de depósito de datos y funciona como un sistema gestor de base de datos (SGBD), por otro lado los clientes suelen ser estaciones de trabajo que solicitan varios servicios al servidor. Ambas partes deben ser conectadas entre sí mediante una red de comunicaciones.

En la figura 2.5.1.1 Se muestra la representación gráfica de la arquitectura Cliente/Servidor.



2.5.1.1 Representación gráfica de la Arquitectura Cliente/Servidor

Este tipo de arquitectura es la más utilizada en la actualidad, debido a que es la más avanzada y la que mejor ha evolucionado en estos últimos años, en general esta arquitectura necesita tres tipos de software para su correcto funcionamiento:

- Software de gestión de datos: Este software se encarga de la manipulación y gestión de los datos almacenados y requeridos por las diferentes aplicaciones, normalmente este software se aloja en el servidor.
- Software de desarrollo: este tipo de software se aloja en los clientes y solo en aquellos que se dedique al desarrollo de aplicaciones.
- Software de interacción con los datos: también reside en los clientes y es la aplicación gráfica de usuarios para la manipulación de datos, siempre a nivel de usuario (consultas principalmente).

2.5.2 Características de la Arquitectura Cliente/Servidor.

- Se establece una relación entre procesos distintos, los cuales pueden ser ejecutados en la misma máquina o en máquinas diferentes, distribuidas a lo largo de la red.
- Existe una clara distinción de funciones basada en el concepto de “servicio”, que se establece entre clientes y servidores.



- La relación establecida puede ser de muchos a uno, en la que un servidor puede dar servicio a muchos clientes, regulando su acceso a recursos compartidos.
- Los clientes corresponden a procesos activos en cuanto a que son estos los que hacen peticiones de servicios a los servidores. Estos últimos tienen un carácter pasivo ya que esperan las peticiones de los clientes.
- Las plataformas de software y hardware entre clientes y servidores son independientes.
- El concepto de escalabilidad tanto horizontal como vertical es aplicable a cualquier sistema cliente/servidor. La escalabilidad horizontal permite agregar más estaciones de trabajo activas sin afectar significativamente el rendimiento. La escalabilidad vertical permite mejorar las características del servidor o agregar múltiples servidores.

Modelo Cliente/Servidor de una capa.

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

Modelo cliente/servidor de dos capas.

Tradicionalmente la arquitectura cliente/servidor está basada en un modelo de dos capas. Este modelo consta de un cliente y un servidor de base de datos. El procesamiento de tareas y la lógica de la aplicación son compartidas entre el servidor de base de datos y el cliente.

Las aplicaciones agrupan la lógica de presentación y la lógica de aplicación en una máquina y acceden a fuentes de datos compartidos a través de una conexión de red que se encuentran en el servidor de datos.



Modelo cliente/servidor de tres capas.

Este es un modelo evolucionado de la arquitectura de dos capas, en el que hay una capa intermedia entre el cliente y el servidor de base de datos. Esta capa consiste de un servidor de aplicaciones que contiene el grueso de la lógica de aplicación. Con esta arquitectura la lógica de la aplicación reside en una sola capa de fácil funcionamiento. El diseño arquitectónico de la capa media puede también ser optimizada en funciones del servidor ya que éste no tiene que contener u hospedar la base de datos.

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

En las aplicaciones cliente/servidor de tres capas, la presentación, la lógica de aplicación y la fuente de datos están conceptualmente separados.

Modelo cliente/servidor de “n” capas.

En estas aplicaciones, la presentación, la lógica de aplicación y la fuente de datos, además de estar conceptualmente separados, se pueden subdividir.

2.5.3 Ventajas de la Arquitectura Cliente/Servidor.

- Recursos centralizados: debido a que el servidor es el centro de la red, puede administrar los recursos que son comunes a todos los usuarios, por ejemplo: una base de datos centralizada se utilizaría para evitar problemas provocados por datos contradictorios y redundantes.
- Seguridad mejorada: ya que la cantidad de puntos de entrada que permite el acceso a los datos no es importante.
- Administración al nivel del servidor, ya que los clientes no juegan un papel importante en este modelo, requieren menos administración.



- Redes escalables, gracias a esta arquitectura, es posible quitar o agregar clientes sin afectar el funcionamiento de la red y sin la necesidad de realizar mayores modificaciones.
- El cliente no necesita tanta potencia y capacidad de procesamiento, ya que parte del proceso se reparte con los servidores.
- El desarrollo y mantenimiento de aplicaciones del modelo cliente/servidor es más rápido, ya que se pueden emplear herramientas existentes, como los servidores de SQL.
- El tráfico de red se reduce de una forma considerable. El cliente se conecta al servidor cuando es estrictamente necesario, obtiene los datos que necesita y cierra la conexión dejando la red libre para otra conexión.
- Contribuye a proporcionar a los diferentes departamentos de una organización, soluciones locales, pero permitiendo la integración de la información relevante a nivel global.
- Facilita la integración entre sistemas diferentes y comparte información, permitiendo, por ejemplo, que las máquinas ya existentes puedan ser utilizadas haciendo uso de interfaces más amigables al usuario. De esta manera, se pueden integrar computadoras personales con sistemas grandes y medianos, sin necesidad de que todos tengan que utilizar el mismo sistema operacional.
- La estructura inherentemente modular, facilita además, la integración de nuevas tecnologías y el crecimiento de la infraestructura computacional, favoreciendo así la escalabilidad de las soluciones.
- Con el favorecimiento del uso de interfaces gráficas interactivas, los sistemas construidos bajo este esquema, tienen mayor interacción intuitiva con el usuario.

2.5.4 Desventajas de la Arquitectura Cliente/Servidor.

- Se cuenta con muy escasas herramientas para la administración y ajuste del desempeño de los sistemas.
- El desempeño se ve afectado por el exceso de tráfico en la red.



-
- Es importante que los servidores y los clientes tengan los mismos mecanismos generales que existan en diferentes plataformas.
 - Las distancias entre los clientes y los servidores contribuye en la respuesta de la información.
 - Se requiere de estrategias para el manejo de errores y para mantener la consistencia de los datos.
 - La complejidad en la seguridad es alta, ya que se deben hacer verificaciones en el cliente y el servidor.



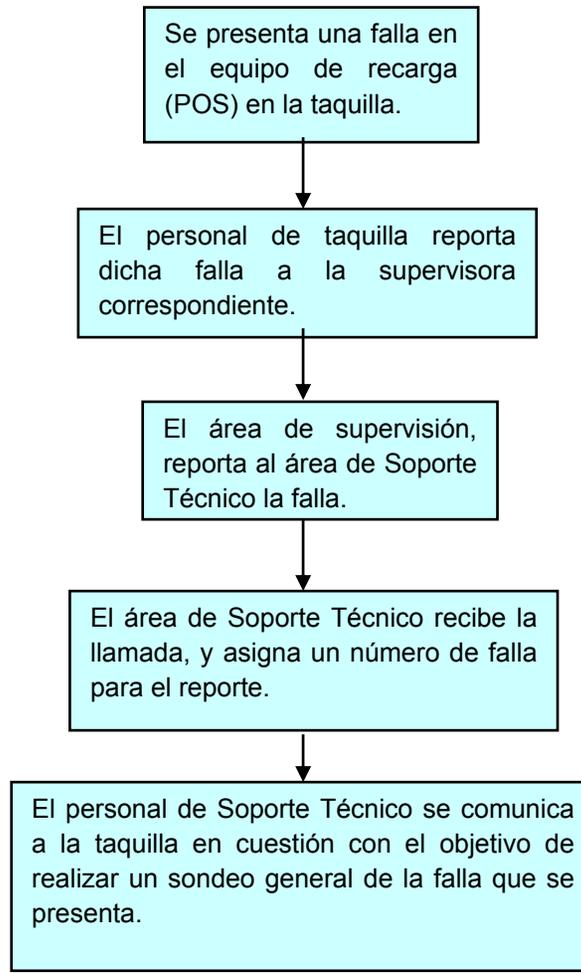
CAPÍTULO 3

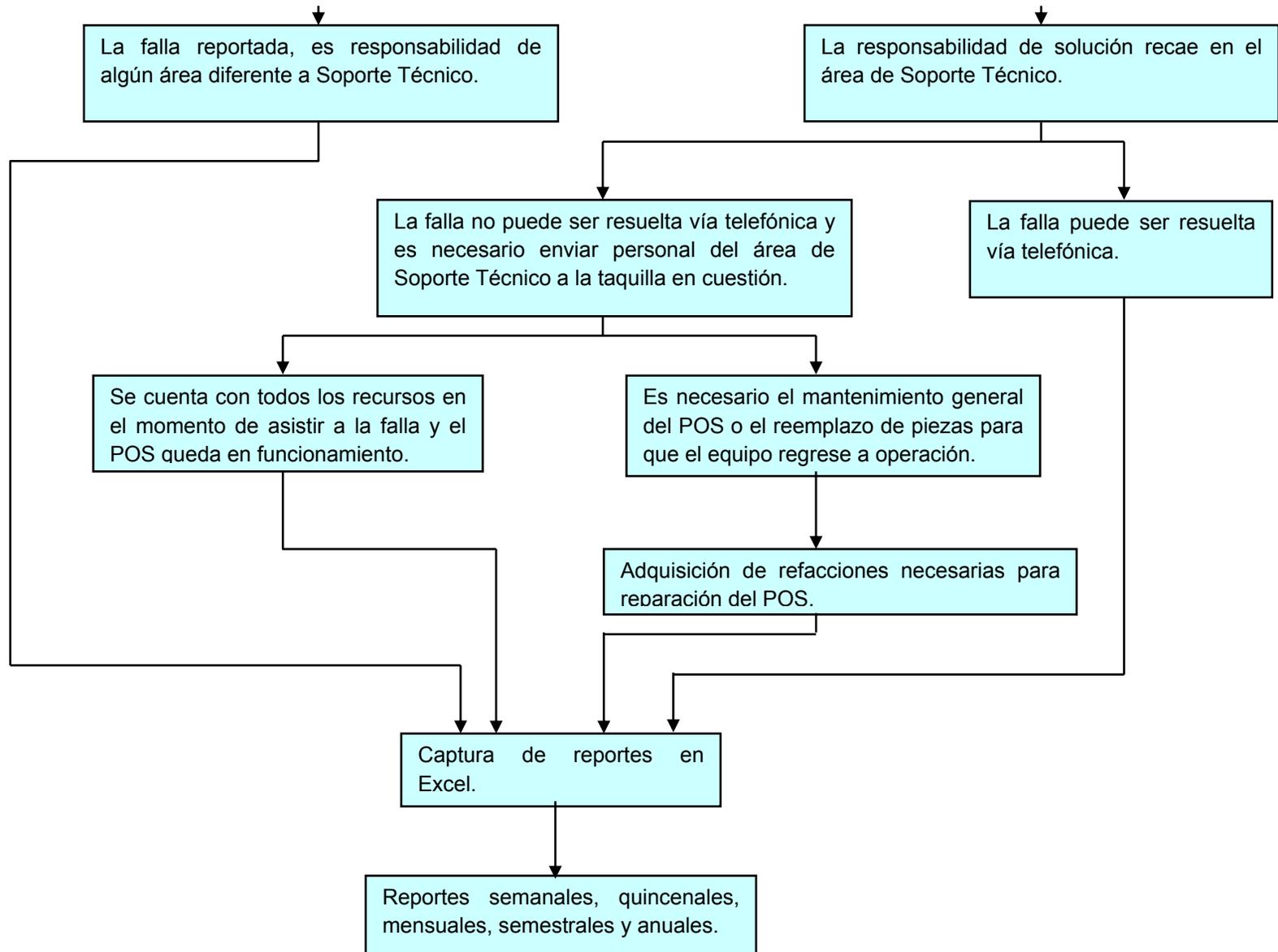


ANÁLISIS Y PLANTEAMIENTO DEL PROBLEMA.

3.1 Problemática Actual.

El siguiente esquema representa a groso modo el flujo de pasos que se deben seguir cuando se presenta alguna falla en cualquiera de las taquillas que conforman la red del metro dentro de sus 175 estaciones.







➤ **Se presenta una falla en el POS:** Lo primero que debe suceder dentro de este proceso es: que se presente una falla en la maquina de recarga dentro de cualquiera de las taquillas que conforman la red de Sistema de Transporte Colectivo, estas fallas pueden ser de diversa índole, a continuación un listado de las más frecuentes:

- ✓ **Módulo CSC fuera de servicio:** Esta falla se presenta cuando el lector que identifica y recarga las tarjetas de prepago para acceso al Sistema de Transporte Colectivo no detecta la presencia de una tarjeta, y en pantalla muestra la leyenda “Modulo CSC no conectado”.
- ✓ **Falla de teclado:** En general, esta situación se presenta cuando al presentar una tarjeta en el CSC y digitar la cantidad o monto a recargar, este no aparece en la ventana correspondiente y por lo tanto la recarga no puede ser efectuada.
- ✓ **Falla de la impresora:** Dentro de las taquillas se ha instalado una impresora térmica que tiene la función de proporcionar un comprobante impreso al personal de taquilla que ampara el monto total que se recargó durante el turno, esto con la finalidad de que dicho personal pueda realizar su corte de caja con la seguridad de no tener faltantes o sobrantes monetarios en su depósito, pero cuando este comprobante sale en blanco de la impresora o simplemente no sale, se reporta la falla de la impresora.
- ✓ **No enciende el equipo:** Esta falla se presenta cuando el POS no enciende, cabe mencionar que es en esta falla donde comúnmente se presenta una situación no imputable al área de Soporte Técnico, es debido a que en ocasiones el equipo no enciende a causa de la ausencia de energía eléctrica en los contactos que alimentan al equipo, esta falla debe ser atendida por el área de Baja Tensión del Sistema de Transporte Colectivo.
- ✓ **Falla del Mouse:** En general es cuando el mouse no “responde” para las funciones requeridas por el personal de taquilla.
- ✓ **Falla del Display:** En las ventanillas de cada una de las taquillas que cuentan con un POS instalado, se encuentra una pantalla que tiene la



leyenda “Bienvenido STC !” cuando no hay una tarjeta presente en el CSC, es decir, durante el tiempo en que el sistema está en espera de una tarjeta, y que de igual forma muestra la cantidad que ha sido recargada al momento en que esta se efectúa, en caso de que las leyendas que este dispositivo debe mostrar no sean las correctas o simplemente no se presente ninguna, se reporta la falla del display.

- ✓ **Falla de recarga:** En este caso, lo que sucede es que por algún motivo la recarga no fue procesada por la tarjeta que fue recibida en taquilla, aunque en la mayoría de los casos esto se debe a un intento de fraude por parte del usuario, en otros casos sí es por causa de falla en alguno de los dispositivos vinculados con el proceso de recarga de una tarjeta, por ejemplo: el lector de tarjetas (CSC).
- ✓ **Pantalla incorrecta:** Esta falla en general se da de alta cuando en el monitor del POS aparece una ventana diferente a la que debiese aparecer en el proceso de recarga de una tarjeta de prepago, en la mayoría de ocasiones se debe a un error del personal de taquilla al seleccionar una ventana errónea y que además no forma parte de las ventanas inmiscuidas en el proceso de recarga, y en otras ocasiones es debido a la aparición de mensajes de error o avisos en general del sistema (Volcados de pila, errores de conexión a la Base de Datos, cuando la tarjeta presentada no es la esperada, etc.).
- ✓ **Equipo trabado:** Se presenta cuando por errores de hardware o de software en el equipo y los dispositivos periféricos que lo integran dejan de “responder” a las peticiones del personal de taquilla.
- ✓ **Bloqueo de cuenta:** Para el acceso al programa de recarga dentro de la taquilla, cada taquillera posee una clave de usuario y un password, por seguridad del propio personal de taquilla esta clave se bloquea al ingresar erróneamente estas claves en tres ocasiones, cabe mencionar que estos intentos no son consecutivos si no acumulativos.



- **El personal de taquilla reporta la falla al área de supervisión:** Debido a la estructura jerárquica que existe en el Sistema de Transporte Colectivo, el personal de taquilla debe reportar todas y cada una de las incidencias que se susciten en la taquilla durante su turno con su supervisora encargada de tramo, la cual a su vez tomara las decisiones prudentes para la solución y el correcto funcionamiento de la taquilla de la cual se trate.

- **El área de supervisión reporta al área de soporte:** Una vez que la supervisora de tramo correspondiente a considerado prudente turnar la incidencia como falla al área de Soporte Técnico, realiza una llamada por medio de las extensiones que han sido designadas para este fin, con el objetivo de levantar un reporte para la atención de dicha falla.

- **El área de Soporte Técnico recibe la llamada, y asigna un número de falla para el reporte:** El personal de Soporte Técnico recibe la llamada de la supervisora, asigna un número de falla consecutivo con la finalidad de controlar la asignación de reportes, en este paso se solicita a la supervisora datos tales como: número de la taquilla que presenta la falla, número de expediente de la supervisora que realiza el reporte y la falla específica que se presenta en la taquilla.

- **El personal de Soporte Técnico se comunica a la taquilla en cuestión con el objetivo de realizar un sondeo general de la falla que se presenta:** Es necesario que el personal de Soporte Técnico se comunique a la taquilla en cuestión por varios motivos tales como:
 - ✓ Verificar que la falla sea la reportada, debido a que en repetidas ocasiones la falla reportada no es en realidad la que se presenta en la taquilla y es en comunicación directa con el personal de taquilla que se confirma la falla real que se presenta.



- ✓ Comprobar si la falla puede ser resuelta vía telefónica o es necesaria la presencia de un técnico en la taquilla para darle solución.

➤ **La falla reportada, es responsabilidad de algún área diferente a Soporte Técnico:** En este caso el personal de Soporte Técnico debe verificar que la falla no se deba a circunstancias ajenas al área, tales como la ausencia de energía eléctrica, caso en el cual la falla es reportada al área de Baja Tensión para obtener una solución adecuada, acto seguido capturar el reporte correspondiente, asentando que la falla fue canalizada y darla por resuelta.

➤ **La responsabilidad de solución recae en el área de Soporte Técnico:** Una vez que se ha corroborado que la falla presente en la taquilla es competencia del área de Soporte Técnico proceder con las posibilidades de solución.

➤ **La falla puede ser resuelta vía telefónica:** Existen fallas en extremo fáciles de resolver, caso en el cual se brinda una asesoría vía telefónica al personal de taquilla para que siguiendo los pasos dictados el equipo logre regresar a una operación normal, las soluciones más comunes son:

- ✓ Reconexión del CSC.
- ✓ Bloque numérico desactivado.
- ✓ No Break apagado.
- ✓ Y las fallas susceptibles de ser resueltas al reiniciar el POS (cuando no reconoce algún dispositivo aunque este se encuentre conectado).

Una vez que se ha resuelto la falla reportada, capturar el reporte correspondiente junto con la solución asentando que ha quedado atendido.

➤ **La falla no puede ser resuelta vía telefónica y es necesario enviar personal del área de Soporte Técnico a la taquilla en cuestión:** En otros casos la solución telefónica no es posible y se hace necesaria la presencia de un técnico



en la taquilla para poder resolver la falla, cabe mencionar que no siempre es posible la solución inmediata de la falla, ya que hay ocasiones en las que se necesitan refacciones o herramientas con las que no se cuenta en el momento.

- ✓ Reconexión de los diversos dispositivos que conforman al POS (Mouse, teclado, display, CSC).
- ✓ Desbloques de cuenta.
- ✓ Papel mal colocado en la impresora.
- ✓ Reconexión del cable de red.
- ✓ Cambio de teclas al teclado cuando las originales ya no se ven.
- ✓ Reconfiguración del BIOS del POS.
- ✓ Mantenimiento a los ventiladores de enfriamiento del POS.

➤ **Se cuenta con todos los recursos en el momento de asistir a la falla y el POS queda en funcionamiento:** En la mayoría de los casos que son reportados al área de Soporte Técnico y que requieren de la asistencia de un técnico a la taquilla, la solución es posible con la herramienta y conocimiento con que se cuenta en el área, las fallas más comunes en este caso son:

- ✓ Reconexión de los diversos dispositivos que conforman al POS (Mouse, teclado, display, CSC).
- ✓ Desbloques de cuenta.
- ✓ Papel mal colocado en la impresora.
- ✓ Reconexión del cable de red.
- ✓ Cambio de teclas al teclado cuando las originales ya no se ven.
- ✓ Reconfiguración del BIOS del POS.
- ✓ Mantenimiento a los ventiladores de enfriamiento del POS.

➤ **Es necesario el mantenimiento general del POS o el reemplazo de piezas para que el equipo regrese a operación.** En algunos casos, es necesario el mantenimiento general del POS, es en estas ocasiones que se debe retirar el



equipo de la taquilla y llevarlo al área de Soporte para poder trabajar en él. Una vez terminado el mantenimiento regresar el POS a taquilla y asentar en el reporte correspondiente que ha sido completada la atención de la falla quedando resuelta.

➤ **Adquisición de refacciones necesarias para reparación del POS:** Hay fallas que puede presentar el POS en las que se han dañado uno o más elementos del hardware que componen a éste equipo, en este caso es retirado el POS de la taquilla, si se cuenta con la refacción o refacciones en el área de Soporte Técnico la reparación es inmediata en caso contrario se realiza la requisición de las refacciones pertinentes al almacén del Sistema de Transporte Colectivo y cuando éste las proporciona al área, el equipo es reparado y regresado a la taquilla correspondiente.

En cualquiera de los dos casos, una vez que el POS a sido reinstalado en la taquilla correspondiente se asienta en el reporte que el equipo ha sido reparado, devuelto a taquilla y que se encuentra en correcto funcionamiento, dando el reporte por atendido.

En este punto es importante mencionar que si no se cuenta con la información adecuada en tiempo y forma, es difícil realizar una proyección a largo plazo de las refacciones y herramientas necesarias para el área y hacer la solicitud correspondiente.

➤ **Captura de reportes en Excel:** El control de reportes en la actualidad se lleva en un archivo generado en Excel, en el siguiente subtema se detallan los inconvenientes de este hecho.

➤ **Reportes semanales, quincenales, mensuales, semestrales y anuales:** A continuación una tabla que muestra los reportes entregados (figura 3.1.1):



	SEMANAL	QUINCENAL	MENSUAL	SEMESTRAL	ANUAL
REPORTE DE ACTIVIDADES	GOS INTERNO	INTERNO	GOS INTERNO	GOS INTERNO	GOS INTERNO
REPORTE POR TIPO DE FALLAS	GOS INTERNO	CT INTERNO	INTERNO	CT INTERNO	CT INTERNO
REPORTE POR TIPO DE SOLUCIONES	GOS INTERNO	CT INTERNO	INTERNO	CT INTERNO	CT INTERNO
REPORTE DE PRODUCTIVIDAD DEL PERSONAL	GOS INTERNO	GOS INTERNO	GOS INTERNO	GOS INTERNO	GOS INTERNO

Figura 3.1.1 Tabla esquemática de los reportes entregados por el área de Soporte Técnico.

NOTA: GOS = Gerencia de Organización y Sistemas.

CT = Coordinación de Taquillas.

En la actualidad, la generación de reportes es una tarea difícil debido a la manera en la que la información se encuentra capturada y almacenada.

3.1.1 Problemas específicos al usar Excel.

El uso de Excel en la captura de datos genera inconvenientes tales como errores de dedo, anomalías en cuanto a la concordancia de datos, no se logra una manipulación de los datos de manera eficiente y tampoco una interfaz agradable para el usuario que logre un buen entendimiento del proceso de captura, se incrementa el tiempo, para llevar a cabo un análisis detallado de toda la información generando dificultad en el momento de generar los reportes correspondientes para las áreas que los solicitan (a nivel interno, la Coordinación de Taquillas y la Gerencia de Organización y Sistemas).

Un aspecto muy importante al momento de capturar los datos son las validaciones, por ejemplo puede que una celda de la hoja electrónica necesite recibir una fecha o un valor monetario para que tenga sentido, y el usuario



introduzca datos de carácter que serían erróneos, esta es una gran limitación que se presenta al trabajar con Excel.

En general las hojas electrónicas son estupendas para que un usuario resuelva sus problemas, aunque son difíciles de usar cuando más de una persona necesita compartir los datos. Las hojas electrónicas son útiles para proporcionar una plantilla para la entrada de datos, pero no realizan bien el trabajo si necesitamos una validación de datos compleja.

Cuando se necesita cambiar una fórmula o el formato de ciertos datos, por lo regular se necesita hacer cambios en varios lugares. Si se desean definir nuevos cálculos sobre datos existentes, se tiene que copiar y modificar un documento existente o crear unos enlaces complejos hacia los archivos que contienen los datos.

Mediante Excel no se logra automatizar las tareas de captura y sobre todo no se logra reducir el error al máximo.

3.2 Requerimientos Generales y Particulares.

Al final del proyecto se desea contar con un sistema que será una solución integral en materia de registro y control de las fallas presentadas en los equipos POS.

3.2.1 Requerimientos Generales.

En base a las necesidades de los usuarios el sistema deberá permitir capturar la información necesaria para registrar la falla de un POS cuando esta sea presentada.

Es necesario tener la facilidad para generar reportes gerenciales que permitan crear informes, o bien estadísticas para las diferentes áreas dentro del organismo cuando esto sea requerido.



Se requiere que para el ingreso al sistema se solicite un usuario y una contraseña, por lo que una vez que se hayan ingresado dichos datos, dará acceso a la pantalla de captura de datos. También debe contar con la opción que permita modificar, crear y eliminar usuarios y contraseñas.

En el diagrama 3.2.1.1 se observan los requerimientos generales del usuario. Así mismo, el sistema deberá permitir al usuario mediante la pantalla principal realizar las siguientes acciones:

- Crear un nuevo registro.
- Guardar los registros nuevos o modificados.
- Modificar un registro previamente realizado.
- Borrar un registro previamente hecho o modificado.
- Buscar algún registro dentro de la Base de datos.
- Imprimir un registro nuevo o modificado.
- Dar de alta datos del personal que reporta, así como del personal que registra y resuelve la falla.
- Dar de alta tipos de falla, y soluciones a las mismas.

En la generación de reportes el usuario tendrá la posibilidad de visualizar e imprimir los reportes estadísticos de solicitudes registradas, atendidas y pendientes que necesite, así como la reimpresión de alguna solicitud en caso de ser necesaria, todo esto, a través de la indicación de los rangos de fechas deseados.

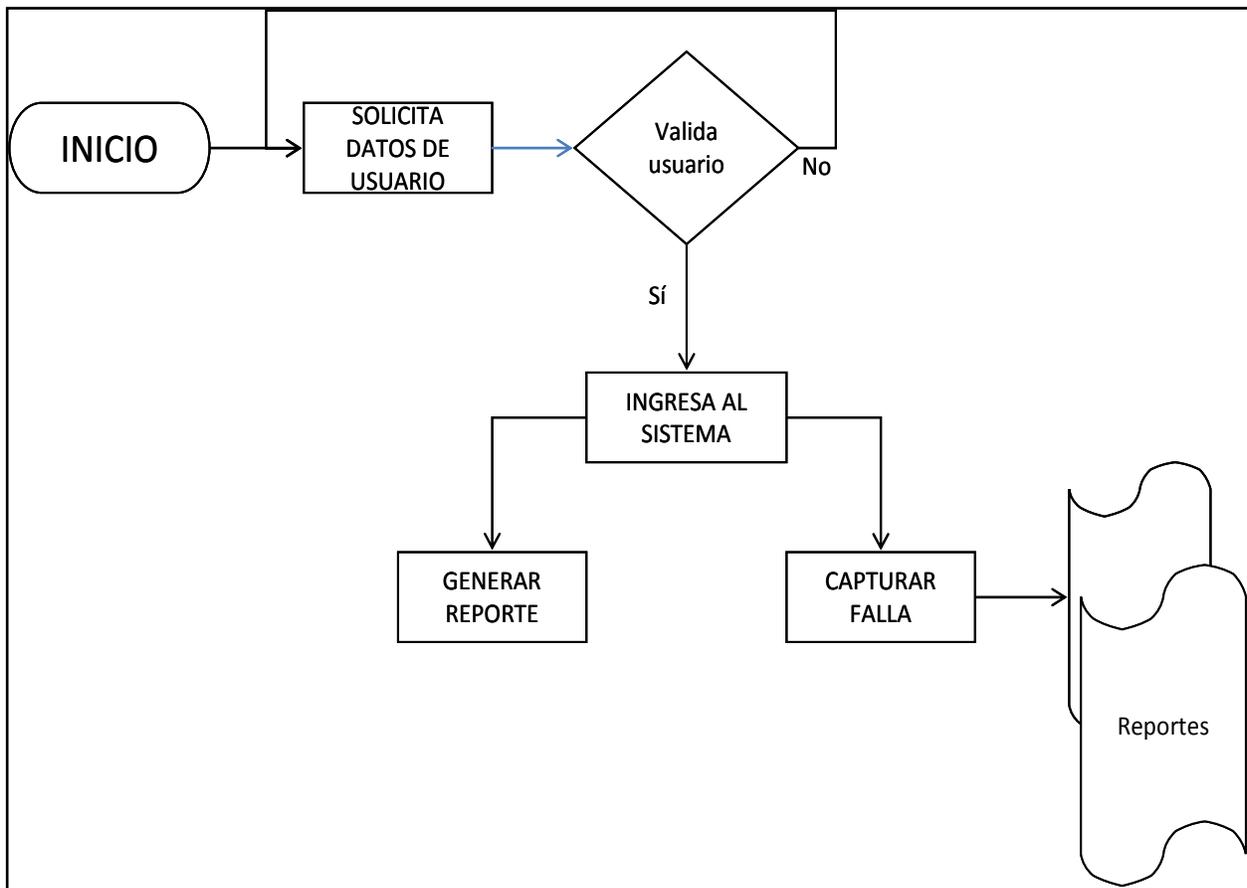


Diagrama 3.2.1.1: Requerimientos generales.

A continuación se enlistan los datos que son necesarios para el registro de una falla reportada en el área.

- Número de solicitud.
- Fecha y hora de la solicitud registrada.
- Expediente, nombre y cargo de quien reporta.
- Estatus de la falla.
- Línea, estación, número de taquilla y número de POS donde se presenta la falla.
- Tipo de falla y solución.
- Nombre y turno del ingeniero que atendió la falla.
- Fecha, hora y tipo de atención de la falla.



Además es importante tener los datos del personal que acredita la solución a una falla reportada anteriormente, esto con la finalidad de garantizar que el reporte registrado ha sido resuelto, los datos necesarios son:

- Expediente, nombre y turno de quien acredita la solución de un reporte.

3.2.2 Requerimientos Particulares.

Debido a las exigencias del área es importante tener en consideración cada uno de los requerimientos particulares del área, con la finalidad de otorgar a los usuarios la información necesaria, integra, detallada y actualizada cuando esta sea requerida.

Por lo cual se requiere que el sistema a desarrollar cuente con una opción en donde el usuario pueda visualizar e imprimir las fallas atendidas.

Dentro de esta opción, el usuario tendrá la oportunidad de:

- Consultar todas las fallas atendidas durante un período de tiempo.
- Consultar las fallas atendidas, eligiendo el turno correspondiente a la persona que reportó la falla y durante un período de tiempo.
- Consultar las fallas atendidas, por el tipo de falla reportada durante un período de tiempo.
- Consultar las fallas atendidas, por la solución que se dio durante un período de tiempo.
- Consultar las fallas atendidas por línea, durante un período de tiempo.
- Consultar las fallas atendidas indicando el turno, tipo de falla, solución y línea durante un período de tiempo.
- Consultar las fallas que se encuentran pendientes de solución.

El área de soporte al POS requiere generar reportes semanales, esto con finalidad de que la GOS pueda llevar un control adecuado sobre las solicitudes que están



siendo atendidas, y estar en la posibilidad de autorizar los estímulos correspondientes para los ingenieros eventuales; así como, comprobar la adquisición de refacciones y con ello obtener la autorización del área correspondiente para hacer las solicitudes de equipo requeridas.

Por tal motivo dentro de la generación de reportes el usuario podrá elegir el rango de fechas de las que desea la información.

Dentro de los reportes que podrá generar tenemos los siguientes:

- **Solicitudes Atendidas:** Este reporte deberá arrojar al usuario una gráfica de las solicitudes atendidas durante un período de tiempo, el cual puede ser durante un día, una semana, una quincena, un mes, un bimestre, un semestre o un año.
- **Solicitudes Atendidas por técnico:** Este reporte deberá arrojar al usuario una gráfica de las solicitudes atendidas por técnico, durante una semana, una quincena, un mes, un bimestre, un semestre o un año. Para así poder el encargado del área brindar los estímulos monetarios a las personas que registren un mayor número de fallas atendidas.
- **Fallas registradas:** Deberá contener la información de las fallas registradas durante el tiempo que el usuario indique, puede ser por día, por semana, por quincena, por mes, por bimestre, por semestre o por año.
- **Fallas registradas por tipo:** Deberá presentar al usuario una gráfica de las fallas por tipo presentadas durante un período de tiempo los cuales pueden ser por día, por semana, por quincena, por mes, por bimestre, por semestre o por año. Con este reporte el usuario podrá identificar las fallas y soluciones que se presentan con mayor frecuencia, y estar en posibilidad de adquirir las refacciones necesarias para resolver las fallas presentadas en el menor tiempo posible.
- **Fallas pendientes:** Podrá imprimir las fallas que se encuentran pendientes por resolver durante un período de tiempo.



Reporte General: Con el fin de mantener informada a la GOS, el área de soporte técnico deberá contar con un informe general pero detallado, el cual deberá contener durante una período de tiempo, las fallas registradas, fallas atendidas, fallas pendientes así como la solución que se le brindó a cada una de ellas.

3.3 Identificación de los posibles módulos del sistema.

En este apartado se hará una breve descripción de los módulos que tentativamente tendrá el sistema vea la figura 3.3.1 y de las funciones que deben cumplir.

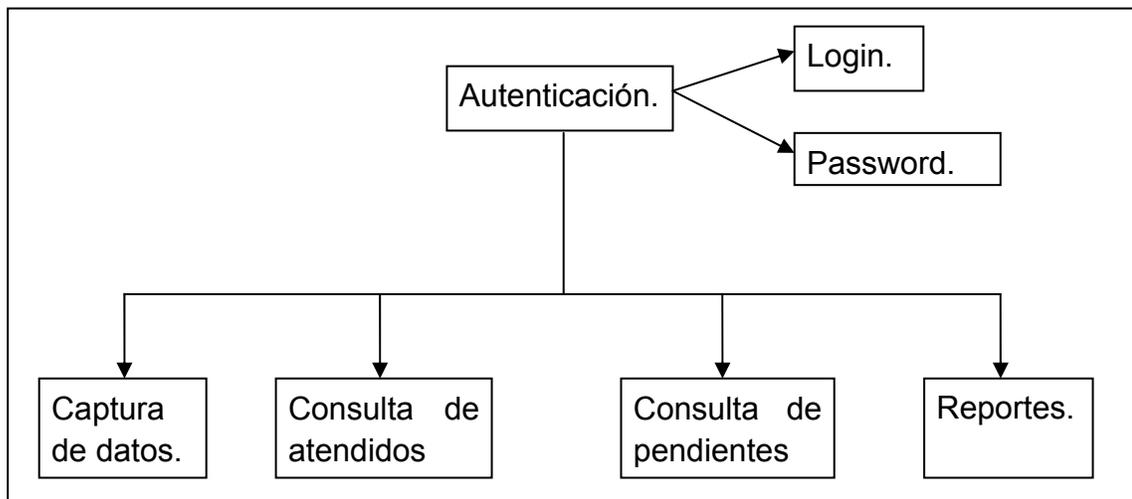


Figura 3.3.1 Módulos tentativos del sistema.

Autenticación: En primera instancia debe existir un módulo de autenticación de usuario, esto con el objetivo de mostrar o no el resto de los módulos del sistema ya que no todos los usuarios tendrán los mismos privilegios ni tampoco acceso a la totalidad de los módulos restantes.

Básicamente existirán dos tipos de usuario del presente, estos son:

- El coordinador del área de Soporte Técnico, con acceso total a los módulos restantes.



- El personal técnico, que solo tendrá acceso al módulo de Captura de Datos y de pendientes.

Captura de datos: Por medio de esta interfaz, el personal técnico del área de Soporte, podrá realizar la captura de los reportes en tiempo real, obteniendo de inmediato un número de falla, que será proporcionado al área de Supervisión de taquilla para un mejor control de las fallas asignadas.

El módulo de captura estará disponible para todos los usuarios del sistema, se sugiere que contenga los siguientes campos:

- Numero de solicitud.
- Fecha de reporte.
- Hora de reporte.
- Estatus.
- Expediente de quien reporta.
- Nombre de quien reporta.
- Cargo.
- Número de taquilla.
- Turno del reporte.
- Número de POS.
- Línea.
- Estación.
- Tipo de falla.
- Solución.
- Nombre de quien recibe el reporte.
- Nombre del técnico que resuelve la falla.
- Turno.
- Tipo de atención.
- Fecha de atención.
- Hora de atención.



- Expediente del personal de taquilla.
- Nombre del personal de taquilla.
- Turno del personal de taquilla.

También deberá presentar los siguientes botones:

- Guardar.
- Limpiar.
- Buscar.
- Imprimir.
- Salir.

Consulta de reportes atendidos por diferentes filtros: Este módulo sólo estará disponible para el coordinador del área de Soporte Técnico, con la finalidad de facilitarle la asignación de recursos humanos y materiales así como la solicitud de refacciones y herramientas en función de observar las fallas o algunos otros parámetros que le permitan tomar las decisiones que considere prudentes, los campos propuestos son:

- Fecha inicial.
- Fecha final.
- Taquilla.
- Turno.
- Tipo de falla.
- Tipo de solución.
- Línea.
- Expediente del personal de taquilla.

Mediante estos campos se podrá filtrar la información obteniendo resultados específicos por medio de un parámetro o una combinación de ellos.



Consulta de pendientes: Por medio de esta parte de la interfaz se podrá consultar en todo momento si existen reportes que no han sido atendidos y el motivo por el cual se encuentran en ese estado con la finalidad de solicitar las refacciones y asignar al personal en el momento que éstas se tengan para la solución de la falla del POS en cuestión, para esta ventana específica no es necesario agregar algún campo ya que siempre mostrará sólo la información requerida para este caso, solo se sugiere implementar por medio de botón “Buscar”, implementar una función que refresque la información en pantalla.

Este módulo se encontrará disponible para todos los usuarios del sistema.

Generador de reportes estadísticos: Por medio de esta interfaz se generaran los reportes estadísticos por medio de gráficas o de tablas que reflejan la productividad del personal del área de Soporte Técnico así como las fallas atendidas entre periodos específicos.

Este módulo solo estará disponible para el coordinador del área, se sugiere tenga las siguientes características:

- Un par de botones de selección: Reporte estadístico de solicitudes y Reimpresión de solicitud.
- Un par de cajas para ingresar la fecha inicial y la fecha final para generar las gráficas de productividad.

En el capítulo 4 se hablará ampliamente de todos los módulos a detalle, así como de las funciones que tendrán cada botón y campo existente en ellos.

3.4 Justificación en la elección de las herramientas de solución.

Para llevar a cabo el desarrollo de nuestro sistema se han elegido diversas aplicaciones las cuales todas son compatibles con el sistema operativo Microsoft Windows.



Para el desarrollo de nuestro back-end se han elegido los Gestores de datos: Microsoft SQL Server 2005, Oracle 10g y ASE (Adaptive Server Enterprise / Servidor Empresarial Adaptable) 12.5 de Sybase. Las características, ventajas y desventajas de Microsoft SQL Server 2005 ya sean mencionado en el capítulo II, inciso 2.2, por lo que en este capítulo solamente se trataran las características, ventajas y desventajas de Oracle 10g y ASE 12.5, así como una tabla comparativa de los tres Gestores de datos.

Para crear el front-end se consideraron las aplicaciones Visual Basic 6.0, Delphi 2005 y PB 9 (Power Builder / Constructor Poderoso), de igual forma se mostrará una tabla comparativa entre las tres aplicaciones, así como las características, ventajas y desventajas de Delphi 2005 y PB 9, ya que las de Visual Basic 6.0, ya has sido descritas en el capítulo II, inciso 2.3.

3.4.1 El back-end.

ASE 12.5.

ASE es un Gestor de Base de datos de alto rendimiento y escalabilidad, con soporte a grandes volúmenes de datos, transacciones, usuarios y bajo costo, que permite: almacenar datos de manera segura, tener acceso a ellos, procesarlos y movilizarlos.

Características.

Entre las principales características de este RDBMS se encuentran las siguientes:

- Soporta plataformas: Windows, Linux, Sun Solaris, IBM AIX, HP-UX- Mac OS y Silicon Graphics IRIX.
- Varios tamaños de páginas lógicas: El comando dataserver permite crear dispositivos master y bases de datos con tamaños de páginas lógicas de 2K, 4K, 8K ó 16K.
- Soporta múltiples protocolos de conectividad como Open Client, ODBC, OLE DB, ADO.NET y JDBC.



- Número de columnas y tamaños de columna: El número máximo de columnas que se pueden crear en una tabla es:
 - ✓ 1,024 para columnas de longitud fija tanto en tablas con bloque all-pages (APL) como tablas sólo de datos (DOL).
 - ✓ 254 para columnas de longitud variable en una tabla APL.
 - ✓ 1,024 para columnas de longitud variable en una tabla DOL.
- Soporta a múltiples herramientas de desarrollo y lenguajes de programación como Power Builder, Visual Basic, Java, C++, PHP, entre otros.
- El tamaño máximo de expresiones, variables y argumentos para procedimientos almacenados es de 16K, para cualquier tamaño de página. Estos datos pueden ser binarios o de carácter. Se pueden insertar variables y literales hasta este tamaño máximo en columnas de texto sin utilizar el comando writetex.
- Asignación de la memoria física total de forma dinámica, es decir, no se necesita reiniciar el servidor para asignar más memoria.
- El caché de datos y de procedimientos se especifican como valores absolutos y no varían a menos que se vuelvan a configurar.
- Permite instalar clases de Java en la base de datos y hacer referencia a estas como tipos de datos SQL y también invocar métodos estáticos de Java directamente desde SQL. Puede ajustar métodos estáticos de Java en nombres SQL y generar procedimientos almacenados o funciones SQLJ que puede utilizar como si fuesen procedimientos almacenados Transac-SQL o funciones incorporadas.
- Permite seleccionar datos sin formato ASE mediante XQL y mostrarlos como documentos XML.
- Admite los operadores unión en las instrucciones select que definen vistas. Por ejemplo se pueden usar para dividir una tabla de gran tamaño en subtablas o para crear vistas que hagan referencia a tablas de Oracle, DB2 e Informix; además de tablas locales.
- Puede ser compatible con un sistema externo de archivos.



- Control de acceso en el nivel de fila: permite al propietario de la base de datos o tabla controlar las filas de una tabla a las que pueden tener acceso los usuarios, en función de su identificación o perfil y los privilegios que tiene el usuario del nivel aplicación. Las reglas de acceso se aplican en operaciones select, update y delete.
- Permite suspensión y reanudación de actualizaciones a la base de datos.
- Compatibilidad con Java.NET.
- Copias de seguridad comprimidas.
- Compatibilidad con Unichar.

Oracle 10g.

Oracle 10g es un RDBMS compatible con Windows con una arquitectura cliente/servidor, potente y altamente eficiente.

Oracle es un sistema gestor de datos relacional de última generación, lo cual quiere decir que está orientado al acceso remoto y redes (internet). Hoy por hoy Oracle se puede implementar en diferentes plataformas: Familia de Microsoft, Unix, Linux, Vms, entre otras. Las arquitecturas en las que se asienta pueden ser: Intel, Alpha, Sparc, Risc; a nivel de procesadores.

Oracle es perfectamente configurable en entornos "OLTP", paralelos, Cluster, e incluso resulta una genial solución a nivel de Datawarehouse.

La plataforma para la que en realidad está desarrollada Oracle, y pensada desde un principio es Unix, ya que es un sistema que soporta la mayoría de la carga de los sistemas a nivel mundial, además es un sistema totalmente configurable.

Para la gestión, Oracle y Unix se complementan de una manera casi inalcanzable para otros sistemas, ya que para su administración, Oracle se amolda y apoya en la potencia de Unix, a la vez que aprovecha al 100% su versatilidad y configurabilidad.



Con respecto a Windows, debemos decir que Oracle ha sido capaz de adaptarse de tal manera que hace firme competencia a la base de datos propia de Microsoft SQL- Server.

El servidor Oracle consta de una instancia y una base de datos. Una instancia en Oracle consiste de una estructura de memoria llamada **SGA** (System Global Area / Área Global del Sistema) y **background process** (procesos de fondo) utilizados por un servidor Oracle para manejo de la base de datos. Solamente se puede abrir una y solo una base de datos en cualquier momento.

Características.

Entre las principales características se encuentran las siguientes:

- **Oracle Partitioning** (Particionamiento de Oracle): mejora el manejo de entorno de datos para las aplicaciones OLTP, **data marts** (datos de mercados) y **data warehouse** (colección de datos, organizados, integrados e historizados) añadiéndole más capacidades de manejo, disponibilidad y rendimiento para tablas e índices de gran tamaño. Permite dividir tablas muy grandes en otras más pequeñas y manejables. Los métodos de particionamiento range, hash, list y composite están soportados.
- **Oracle Advanced Security** (Seguridad Avanzada de Oracle): Proporciona una encriptación transparente de los datos almacenados y una encriptación de red para los datos que están viajando a través de la red.
- **Oracle Database Vault** (Bóveda de la Base de Datos de Oracle): Proporciona el control sobre quien, cuando y donde los datos y las aplicaciones pueden ser accedados.
- **Oracle OLAP**: Es un escalable motor de alto rendimiento de cálculo con todo el manejo y administración integrado para la obtención de aplicaciones analíticas. Provee un completo conjunto de funciones analíticas.
- **Oracle Data Mining** (Minería de Datos de Oracle): Permite a las empresas construir avanzadas aplicaciones de inteligencia de negocios a través del



Gestor de base de datos dentro de ella e integrar esta información dentro de las aplicaciones de negocios. Incluye funciones de minería de datos para hacer clasificaciones, predicciones y asociaciones.

- **Oracle Spatial** (Espacial Oracle): Permite a los usuarios y a los desarrolladores de aplicaciones integrar sin dificultades los datos espaciales dentro de las aplicaciones empresariales. Facilita el análisis basado en la relación de datos espaciales, como la proximidad de direcciones de tiendas para clientes en una distancia determinada e ingresos de ventas por territorios.
- **Oracle Tuning Pack** (Paquete de Puesta a punto de Oracle): Provee a los DBA con manejadores expertos de rendimiento para el entorno de Oracle, incluyendo SQL tuning y optimización de almacenamiento. El Tuning Pack incluye: **SQL Tuning Advisor** (Consejero de Optimización de Consultas), **SQL Access Advisor** (Consejero de Accesos SQL) y **Segment Advisor** (Consejero de Segmentos).
- Cuenta con capacidades avanzadas de administración automática y una nueva consola denominada Database Control, a través de la cual se puede monitorear la base de datos en forma proactiva y recibir alertas y consejos para asegurar un desempeño y confiabilidad óptimos.
- Funciona de forma eficaz tanto en pequeños servidores como en servidores **SMP** (Symmetric MultiProcessing / Multiprocesamiento Simétrico), pasando por entornos de red y redes de servidores empresariales.
- Garantiza la integridad de los datos, llevando un control de los registros que han sido modificados dentro de una transacción para poder regresar al estado anterior en caso de que el proceso no finalice completamente.
- Incluye una herramienta para el diseño de pantallas de ingreso, modificación y consulta de registros.
- Soporta una gran cantidad de usuarios concurrentes, ejecutando una gran variedad de aplicaciones de bases de datos.
- Compatibilidad con Microsoft Windows, Unix y Macintosh.



En la tabla 3.4.1.1 se muestran las características de SQL Server 2005 en comparación con los Gestores de datos ASE 12.5 y Oracle 10g.

CARACTERISTICA	SQL SERVER 2005	ASE 12.5	ORACLE 10g
Compatible con Windows XP	◆◆◆	◆◆	◆◆
Facilidad de uso, instalación y aprendizaje	◆◆◆	◆◆	◆◆
Desempeño en búsquedas	◆◆◆	◆◆	◆◆
Soporte XML	◆◆◆	◆◆	◆◆◆
Protocolo SSL	◆◆◆	◆◆	◆◆◆
Funciones SQLJ	◆◆	◆◆◆	◆◆◆
Utilidad de depuración SQL	◆◆◆	◆◆◆	◆◆◆
Compatibilidad con sistema externo de archivos	◆◆	◆◆◆	◆◆
Compatibilidad con Java.NET	◆◆◆	◆◆	◆◆

◆◆◆ Excelente ◆◆ Bueno

Tabla 3.4.1.1 Comparación de los Gestores de la Base de Datos.



3.4.2 El Front-end.

Delphi 2005.

Es una herramienta **RAD** (Rapid Application Development / Desarrollo Rápido de Aplicaciones) diseñado para la programación de propósito general con énfasis en la programación visual. Está basado en una versión moderna de Pascal, denominada Object Pascal. Combina soporte Win32, .NET, Delphi y C++ en un único entorno, permite al equipo y al desarrollador mejorar su productividad y se integra con las soluciones de gestión del ciclo de vida de las aplicaciones de Borland.

Una de las principales características y ventajas es su capacidad para desarrollar aplicaciones con conectividad a base de datos de diferentes fabricantes. El programador cuenta con una gran cantidad de componentes para realizar la conexión, manipulación, presentación y captura de los datos, algunos de ellos liberado bajo licencias de código abierto o gratuito. Estos componentes de acceso a datos pueden enlazarse a una gran variedad de controles visuales, aprovechando las características del lenguaje orientado a objetos, gracias al polimorfismo.

Características.

- Soporta múltiples lenguajes y los SDK que requiere Windows para su desarrollo.
- Es la única solución de Windows que realmente soporta el desarrollo nativo de Win32 y .NET desde la misma herramienta y el mismo lenguaje.
- También soporta ASP.NET, ADO.NET, VCL.NET, y VCL para Win32.
- Dispone de una excelente integración con soluciones ALM. Según la versión de Delphi 2005, se incluye acceso a StarTeam, CaliberRM, Janeva, Optimizetl y ECO.
- Marcadores persistentes: Se trata de marcas especiales que se colocan en el código para simplificar su navegación por el mismo.



- También incluye utilidades para elaborar pruebas unitarias integradas mediante los marcos de pruebas Dunit y Nunit.
- Mejorado soporte BDP ADO.NET para InterBase, Oracle, SQL Server y Sybase.

Power Builder 9.

Es una herramienta de desarrollo empresarial orientada a objetos que permite construir diferentes tipos de aplicaciones y componentes. Se pueden desarrollar aplicaciones cliente/servidor, aplicaciones distribuidas y aplicaciones para Internet.

Power Builder 9 como herramienta profesional de desarrollo de software, es una de las pioneras en el desarrollo gráfico orientado a objetos, permitiendo herencia, encapsulación y polimorfismo haciendo de este ambiente de desarrollo, una excelente herramienta de desarrollo rápido de aplicaciones. Gracias a su madurez, estabilidad y rapidez en el acceso a datos, las principales casas de desarrollo en el mundo así como las grandes empresas prefieren Power Builder para hacer sus grandes desarrollos.

Gracias a sus funcionalidades orientadas a objetos, Power Builder se ha posicionado en el mercado con fama de gran estabilidad y desempeño en tiempo mínimo, ayudando a reutilizar el código y minimizar el impacto en los cambios de las aplicaciones. Además, tiene un excelente desempeño en el manejo de información con Bases de datos empresariales como Oracle, Sybase o Informix, al no accederlas por ODBC sino por driver nativos, haciendo consultas un 25% más rápido. Una de las principales ventajas de Power Builder es la facilidad de trabajar conexiones y actualizar información de bases de datos a través de su objeto datawindow, el cual brinda una de las mayores ventajas competitivas en el mercado.

En la tabla 3.4.2.1 se muestran las características de Visual Basic 6.0® en comparación con las herramientas Delphi 2005 y Power Builder 9, por las que se decidió utilizarlo para la creación del Fron-end.



CARACTERISTICA	Visual Basic 6.0	Delphi 2005	Power Builder 9
Facilidad de aprendizaje y codificación	◆◆◆	◆◆	◆◆
POO	◆◆	◆◆◆	◆◆◆
IDE	◆◆◆	◆◆	◆◆
Objetos de base de datos	◆◆	◆◆◆	◆◆◆
Liberación de memoria	◆◆◆	◆◆	◆◆
Acceso a base de datos	◆◆◆	◆◆	◆◆◆
Manejo de APIs	◆◆◆	◆◆◆	◆◆
Rendimiento	◆◆	◆◆◆	◆◆
Programación pesada	◆◆	◆◆◆	◆◆
Optimización de Código	◆◆	◆◆◆	◆◆
Recursos de terceros	◆◆	◆◆◆	◆◆
Obsolencia	◆◆	◆◆◆	◆◆
Soporte ODBC	◆◆◆	◆◆◆	◆◆

◆◆◆ Excelente ◆◆ Bueno

Tabla 3.4.1.1 Comparación de herramientas para creación del Front-end.



3.4.3 El Reporteador.

Crystal Reports es el generador de Reportes de Visual Basic y con el diseñaremos los reportes de nuestro sistema. Una de las principales ventajas por la cual utilizaremos **Crystal Reports** es que utiliza una interface gráfica a partir de la cual podemos construir cualquier reporte que necesitemos.

A continuación se explicarán los pasos para la creación de un reporte en Crystal Reports.

CREANDO UN NUEVO REPORTE.

Vamos a generar un reporte basado en una tabla de nombre *Agenda* que se encuentra en una Base de Datos de nombre *Controle*. Nuestro reporte deberá obedecer a los siguientes parámetros:

- Campos a ser impresos: *Sobrenombre, Dirección y Fecha de Nacimiento.*
- El reporte deberá ser ordenado por campo *Sobrenombre.*
- Debemos permitir inicialmente la visualización del reporte antes de imprimirlo.
- El nombre del reporte será *Agenda.rpt.*

Seleccionando la opción **New** del menú **File** tendremos el cuadro de la Figura 3.4.3.1 de abajo.

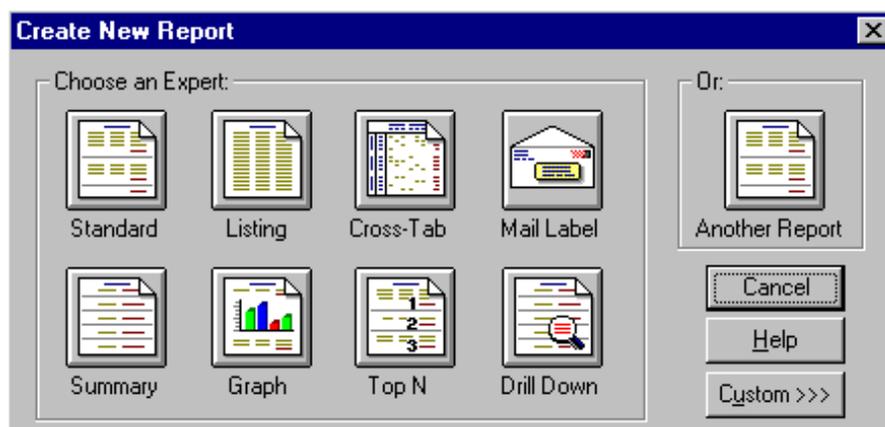


Figura 3.4.3.1 Creación de un nuevo Reporte.



Después de seleccionar el botón **Standard**, debemos seleccionar la Base de Datos en la opción **Data File**, para nuestro caso *Controle.mdb*.

Al continuar tenemos una lista de todas las tablas y consultas grabadas en la Base de Datos. Nosotros trabajaremos con la tabla *Agenda*, por lo que la debemos seleccionar y hacer clic en el botón **Next** para proseguir.

Como nuestro reporte está basado solamente en la tabla *Agenda*, el próximo paso **Links** puede ser pasado por alto, por lo tanto haga clic nuevamente en el botón **Next**.

Vamos a seleccionar los campos de la tabla que serán impresos en el reporte. Seleccione cada campo y haga clic en el botón **Add**, como se muestra en la figura 3.4.3.2.

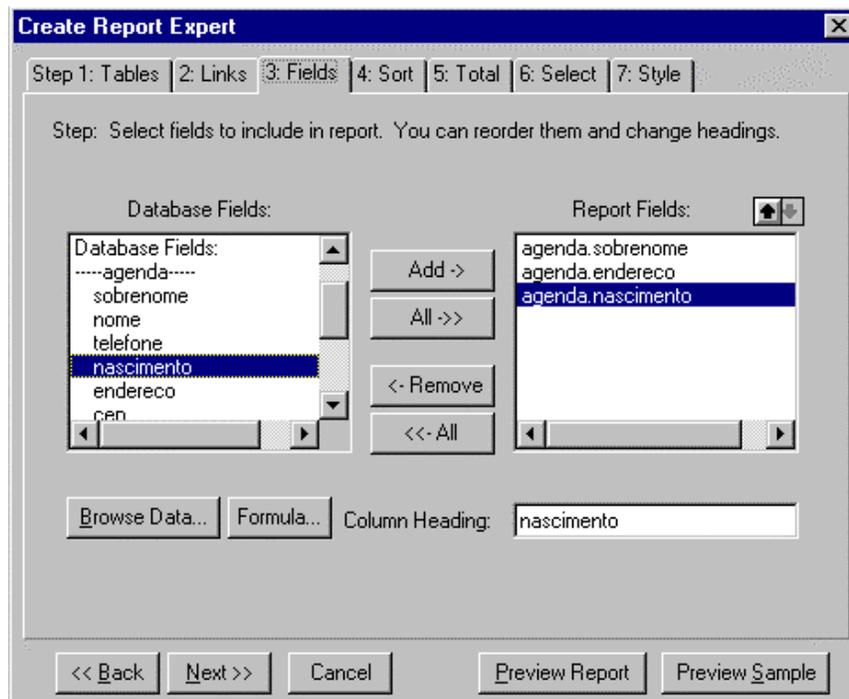


Figura 3.4.3.2 Selección de campos para reporte.



A esta altura el reporte está prácticamente terminado, para visualizarlo se debe dar clic en el botón **Preview Report.**, en donde después de dar clic en la pestaña **Desing** podemos notar cinco secciones. (figura 3.4.3.3).

- **Title:** para el título de la aplicación.
- **Page Header:** contiene los elementos de encabezado de página.
- **Details:** contiene los campos a ser impresos.
- **Page Footer:** se refiere a pie de página.
- **Summary:** impresión de resúmenes.

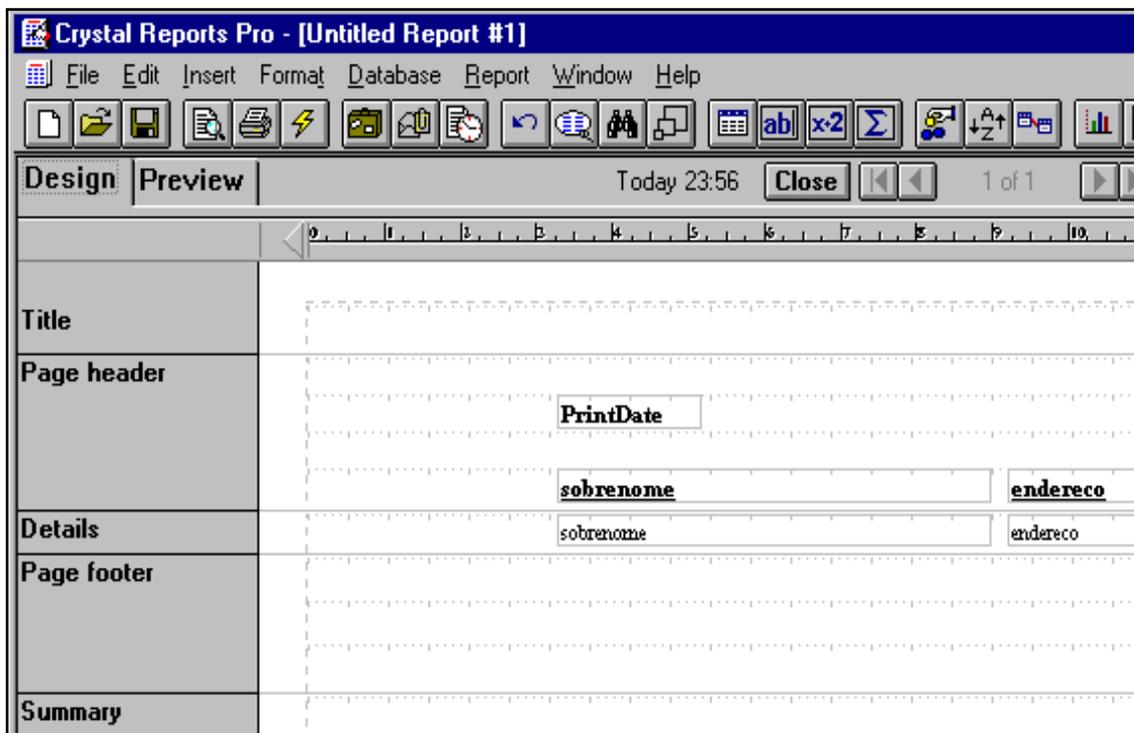


Figura 3.4.3.3 Visualización del reporte.

AGRUPANDO Y ORDENANDO RESUMENES.

Para agrupar registros, ordenándolos por una determinada columna debemos seleccionar la opción **Group Section** del menú **Insert**. Agrupando por el campo *Sobrenombre* en orden ascendente. Vea figura 3.4.3.4.

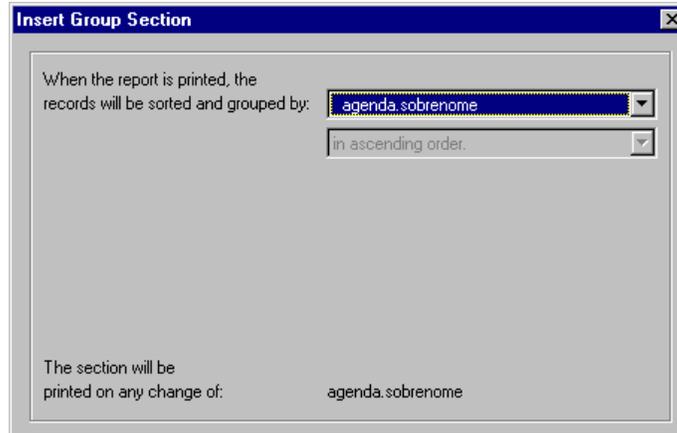


Figura 3.4.3.4 Ordenamiento de forma ascendente.

INSERTANDO TITULOS Y LEYENDAS.

Para insertar un título a nuestro reporte y una leyenda para el campo *Sobrenombre*, debemos seleccionar la opción **Text Field** del menú **Insert**. En la caja de diálogo **Enter Text** debemos escribir el título; en este caso: *Agenda Personal* y hacer clic en el botón **Accept**. Al lado del puntero del mouse hay un rectángulo que se deberá posicionar en el lugar deseado, para este caso, en la sección **Title**.

Para crear la leyenda *Nombre* para el campo *Sobrenombre* como encabezado de grupo, seleccionamos **Text Field** nuevamente, y para este caso le llamaremos *Nombre*, dando clic en **Accept** y posicionando la leyenda en el mismo lugar de la leyenda *Sobrenombre*.

DAR FORMATO A CAMPOS, CAMPOS ESPECIALES Y DISEÑO DE LINEAS.

Para dar formato a los campos basta seleccionar el campo deseado y dar clic en la opción **Format** o haciendo clic en el botón derecho del mouse sobre el campo aparecerá un menú **pop-up**. Vea figura 3.4.3.5.



Figura 3.4.3.5 Menú Pop-up.

Por este menú podemos acceder a las opciones pertinentes de un determinado campo del reporte. Para nuestro caso seleccionamos el título *Agenda Personal*, y vamos a modificar la fuente (**Change Font...**) para un tamaño 14 y estilo negrita. Para modificar más de un campo los seleccionamos manteniendo apretada la tecla **Shift**.

De igual forma podemos insertar diversos campos con diversos formatos, para este ejemplo insertaremos un campo referente a la Fecha en la esquina superior izquierda. Seleccionamos la opción **Special Field** del menú **Insert** y escogemos la opción **Print Date** posicionándolo en el lugar indicado.

FORMULAS.

Crystal Reports nos permite la utilización de fórmulas, en este caso vamos a insertar una fórmula para imprimir el *Número de página* en el pie de página del reporte. Para eso usamos el editor de fórmulas de **Crystal Reports** que puede ser abierto a través del icono (x2) o de la opción **Formula Field** del menú **Insert**.



Después debemos escribir el nombre de la fórmula en el campo **Formula Field**, en este caso escribiremos “*página*” y demos clic en el botón **OK**. Ahora escribiremos la fórmula en la caja **Formula Text**:

“Página: “ + seleccionar la función **TrimLeft** en la lista de Funciones, en la misma lista seleccionar la función **ToText** y finalmente seleccionar el ítem **Page Number** en el final de la lista **Functions**. Al final debemos tener lo siguiente en la caja **Formula Text**:

“Página: “ + TrimLeft (ToText(PageNumber,0))

La función **PageNumber** retorna un valor numérico del número de página por eso usamos la función **ToText** para convertirla en un **String**, y se usa la función **TrimLeft** para remover los espacios en blanco a la derecha.

Para verificar la fórmula debemos dar clic en el botón **Check**, si todo es correcto **Crystal Reports** informará con el mensaje **No errors found** indicando que la sintaxis está correcta.

Ahora basta hacer clic en el botón **Accept** y posicionar la fórmula en la izquierda de la sección **Page Footer**.

Terminado el reporte basta guardarlo a través de la opción **Save** del menú **File** y escribir el nombre para el reporte (nuestro caso escriba *Agenda*).

DETERMINANDO EL ESTILO E INSERTANDO UNA FIGURA EN SU REPORTE.

Para dar un estilo predeterminado a nuestro reporte se puede utilizar la guía **Style** para escoger una forma de presentación de nuestro reporte. Para esto debemos seleccionar uno de los estilos de la caja de lista **Style** y a la derecha se mostrará la vista del reporte.



De igual forma nos permite insertar una figura en el reporte dando clic el botón **Add Picture**.

La caja de texto **Title** nos permite insertar un título para el reporte, como lo muestra la figura 3.4.3.6.

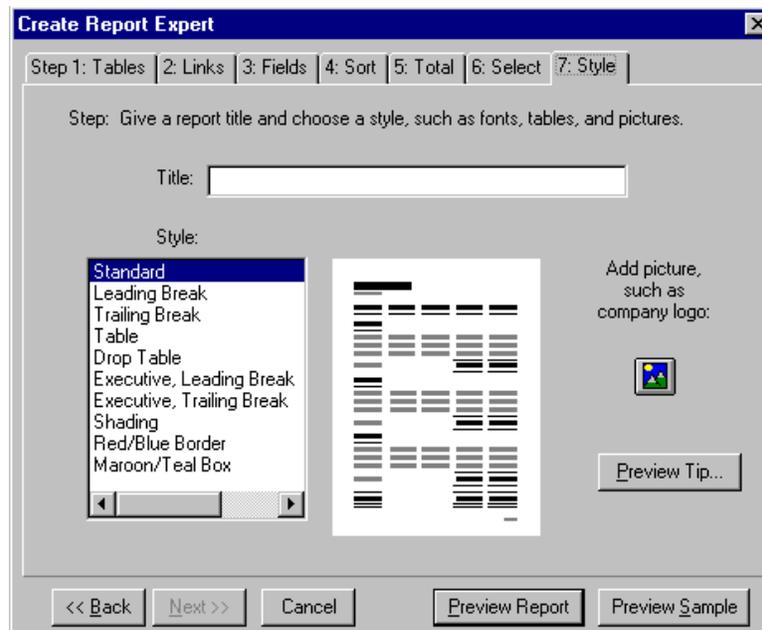


Figura 3.4.3.6 Pantalla guía Style.

IMPRIENDO UN REPORTE A PARTIR DE LA APLICACION VISUAL BASIC 6.0®.

Para poder imprimir nuestro reporte desde Visual Basic 6.0 debemos activar el componente de **Crystal Reports** para nuestra aplicación con la opción **Components** del menú **Projects** y a continuación seleccionar el control **Crystal Reports** y copiarlo para nuestro formulario.

De igual forma se deben definir algunas propiedades para el control **Crystal Reports**.

- **CopiesToPrinter**: Determina el número de copias del reporte. Imprimiremos una (1).



- **Destination:** Direcciona la impresión. En el cuadro 1- Para impresora 2-A un archivo. Escribiremos cero (0).
- **ReportFileName:** Indica la localización del reporte (archivo.RPT) a ser impreso.
- **WindowTitle:** Título de la ventana **Preview**. En este caso escribiremos Agenda.
- **SortFields:** Configura el orden de ordenación.

Finalmente se tiene que crear un botón de comando en el formulario que irá a disparar la impresión del reporte, le llamaremos “Imprime” se le deberá asociar el siguiente código:

Private Sub Imprime_Click()

CrystalReport1.Destination = 0

CrystalReport1.ReportFileName = “C:\Controle\Agenda.rpt”

CrystalReport1.SortFields(0) = “+{Agenda.Sobrenombre}”

CrystalReport1.Action = 1

End Sub

La propiedad **Action** definida como 1 ejecuta la impresión del reporte.

Otra forma de ordenar los registros por código es utilizar la propiedad **SortFields**.

La propiedad **SortFiled**s, “+{Agenda.Sobrenombre}”, indica que el orden de impresión será por campo *Sobrenombre* (Agenda.Sobrenombre) en un orden ascendente (+).





CAPÍTULO 4



DISEÑO Y CONSTRUCCIÓN DEL SISTEMA.

4.1 Elección de la metodología de desarrollo.

El presente trabajo de tesis está basado en la metodología de Análisis Estructurado también conocida como “Top – Down”, y consiste en dividir un problema grande y complejo en componentes más y más pequeños hasta que ya no sea posible realizar más divisiones sobre dicho problema y especificar las relaciones existentes entre los componentes resultantes.

En el capítulo 2 del presente trabajo ya se hablo de diferentes metodologías tales como SSADM y MERISE en comparación con el elegido conocido como YOURDON/De Marco.

La elección de la metodología YOURDON/De Marco está principalmente sustentada en que ésta se haya orientada a procesos, concentrándose en las funciones del sistema requerido. Se basa en la utilización de modelos del sistema que representan procesos, flujos y estructuras de datos de una manera descendente, además existe una transición de una visión general del sistema a un nivel de abstracción más sencillo, desde el nivel 0 del Diagrama Conceptual hasta su mínimo nivel de Diagramas de estructura de las funciones del sistema de acuerdo a la teoría del Diseño estructurado de Yourdon - De Marco.

Se apoya en la realización de técnicas gráficas tales como:

- Diagramas de flujo de datos (D.F.D): Diagrama que representa los flujos o funciones que lleva a cabo el sistema en diferentes niveles y los datos que fluyen entre las mismas funciones.
- Diagramas de Flujo de Procesos (D.F.P): Es una representación gráfica de los pasos que seguimos para realizar un proceso; partiendo de una entrada, y después de realizar una serie de acciones, llegamos a un resultado, entre sus características tenemos: Presenta información clara, ordenada y concisa del flujo que sigue un proceso, está formado por una serie de



símbolos unidos por flechas, cada símbolo representa una acción específica y las flechas entre los símbolos representan el orden en que se realizan las acciones.

- Diccionario de datos: Son las definiciones de todos los datos que aparecen en el diagrama de flujo de datos y procesos.
- Especificaciones de procesos: Definen la obtención de salidas del proceso a través de sus entradas.
- Diagrama entidad /relación.
- Normalización.

4.2 Diagramación.

4.2.1 Diagrama de Contexto.

Se conoce como Diagrama de Contexto de nivel 0. Este es un modelo lógico-gráfico para representar el funcionamiento de un sistema en un proyecto de software.

Sus elementos gráficos según la notación de Yourdon/De Marco se muestran en la tabla 4.2.1.1.

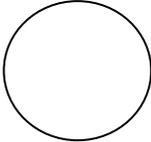
REPRESENTACIÓN GRÁFICA	NOMBRE	DESCRIPCIÓN
	Procesos	Conjunto de actividades de negocio que explican que se hace y como se lleva a cabo.
	Entidad Externa	Persona, grupo de personas o unidad de negocio que entrega y/o recibe información.
	Flujo de datos	Señala el flujo de datos de una entidad externa a un proceso y viceversa, de un proceso a otro, y de un proceso a un almacén de datos y viceversa.
	Almacén de datos	Lugar físico donde se almacenan los datos procesados o desde donde se recuperan para apoyar un proceso.

Tabla 4.2.1.1 Elementos de un DFP.

En un DFP los flujos, entidades externas y almacenes se etiquetan con un nombre. Los procesos se etiquetan con un número y un verbo en infinitivo con objeto directo.

Un DFP puede ser profundizado expandiendo algunos de sus procesos en subprocesos, en este caso la etiqueta tendrá un número adicional. No hay un límite para el número de procesos.

Los diagramas derivados de los procesos principales se clasifican en niveles, los cuales son:



- Nivel 0: Diagrama de contexto.
- Nivel 1: Diagrama de nivel superior.
- Nivel 2: Diagrama de detalle o expansión.

En general, el diagrama de contexto representa una sola burbuja o proceso que identifica la función principal, con el flujo de información de entrada y salida representadas por flechas, que lo relacionan con otros sistemas y personas.

Este diagrama resume el requisito principal del sistema; recibir entradas, procesarlas de acuerdo con una demanda, generar una función y entregar salidas. La figura 4.2.1.2 muestra el diagrama de contexto del Sistema para el registro y control de fallas técnicas de los equipos de recarga (POS) instalados en las taquillas del Sistema de Transporte Colectivo, (SIREPOS).

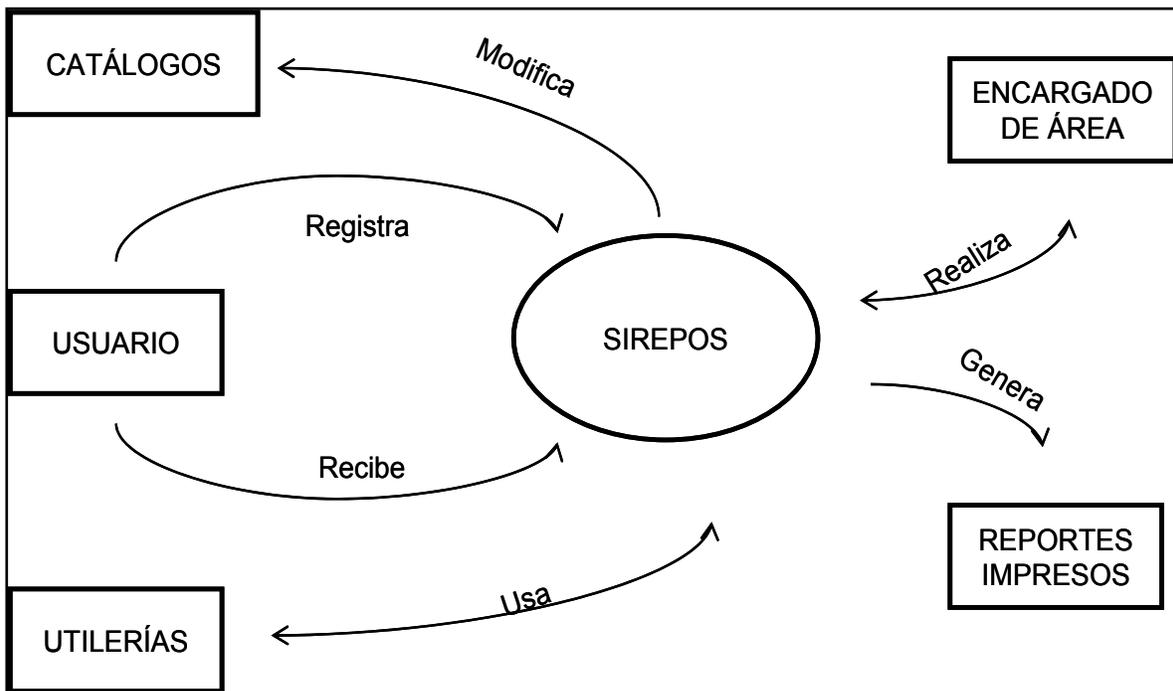


Figura 4.2.1.2 Diagrama de Contexto.

El diagrama de contexto (Vea figura 4.2.1.2) está conformado por los siguientes elementos:



- Usuario que son los trabajadores del área de Soporte Técnico al POS, los cuales son los encargados de registrar cada una de las fallas presentadas en los equipos POS, a través de un nombre de usuario y contraseña.
- El sistema cuenta con diversos catálogos, los cuales pueden ser modificados según se requiera.
- El SIREPOS genera reportes ya sea en pantalla, a un archivo, a una impresora o a una hoja de Excel.
- Dentro de las utilerías que utiliza el sistema se encuentra la importación de datos desde Excel.
- El encargado del área interactúa con el sistema para realizar altas, bajas o cambios en los datos de los trabajadores del área.

4.2.2 Diagrama de flujo de procesos.

Los diagramas de flujo de procesos son un tipo de herramienta de modelado, permiten modelar todo tipo de sistemas, concentrándose en las funciones que realiza, y los datos de entrada y salida de esas funciones.

Un sistema puede representarse empleando varios diagramas de flujos, cada flujo de datos puede representar una parte “más pequeña” del sistema. Los DFP permiten una partición por niveles del sistema.

Los DFP suelen servir para comprender fácilmente el funcionamiento de un sistema.

Como ya se mencionó en el punto anterior, existen tres niveles para un DFP, a continuación se presentarán los DFP de nivel superior y de detalle.

La figura 4.2.2.1 muestra el diagrama de procesos general para nuestro sistema a desarrollar.

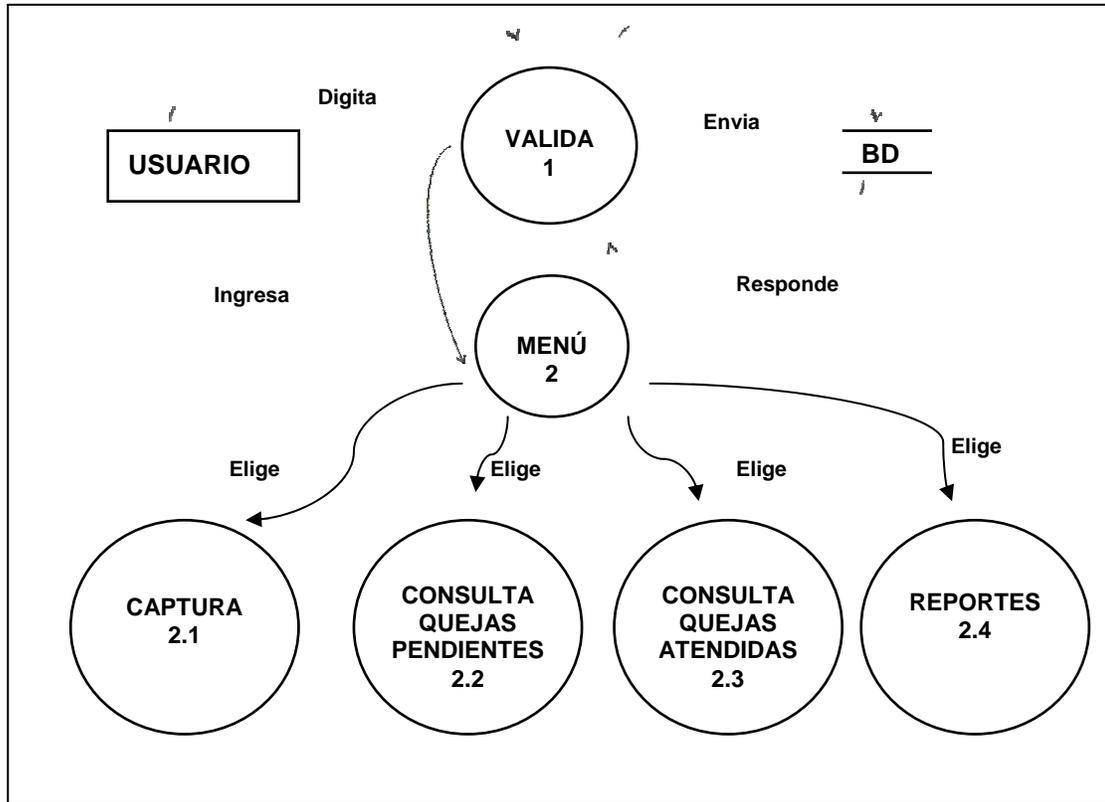


Figura 4.2.2.1 Diagrama de procesos del Sistema.

Para poder acceder a nuestro Sistema se deberá ingresar un usuario y una contraseña asignada, de la cual dependen los privilegios para el sistema, si este usuario y contraseña es correcta y es de administrador, se accederá al menú principal donde se podrá elegir alguno de los cuatro módulos del sistema; de lo contrario, se accederá automáticamente al modulo de captura, en caso de que el usuario y contraseña sean incorrectos el sistema dará oportunidad de volver a digitarlos. (Vea figura 4.2.2.2).

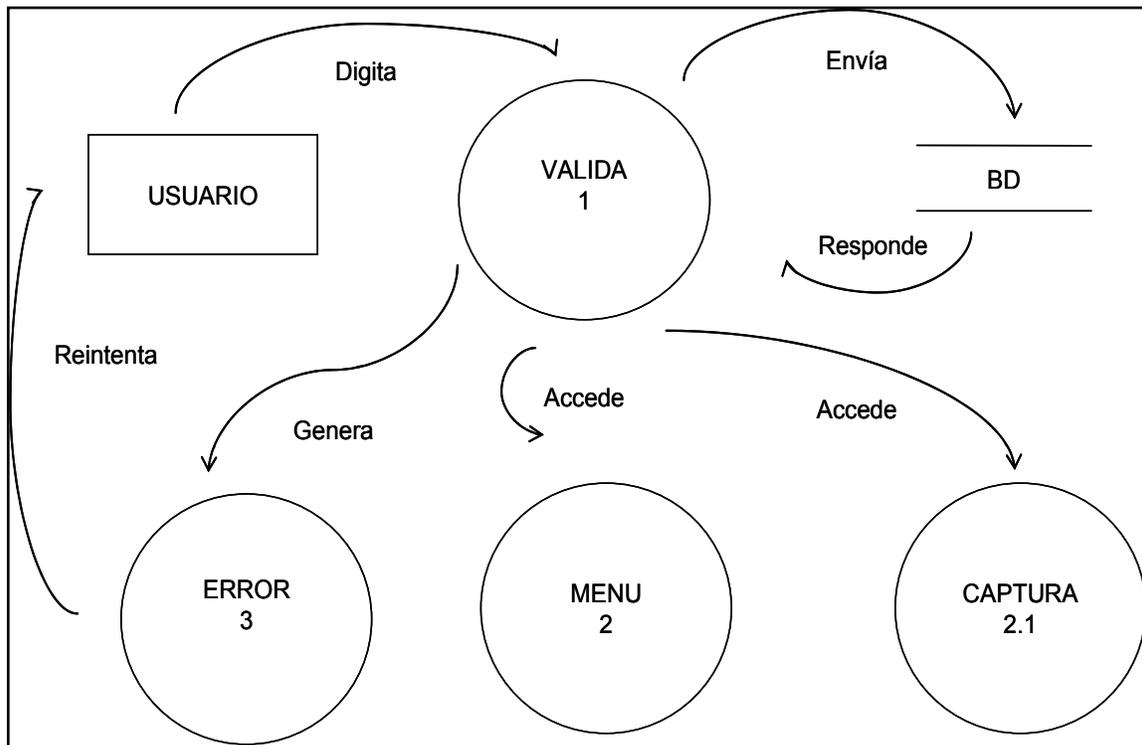


Figura 4.2.2.2 Diagrama de procesos de nivel 2 del sistema.

Una vez que el usuario ha ingresado al sistema con un usuario y contraseña de administrador podrá elegir alguno de los 4 módulos (Vea figura 4.2.2.3), a continuación se muestran los diagramas de procesos para cada uno de los módulos.

En el módulo de Captura de datos, el administrador, puede realizar altas, bajas y cambios de los catálogos que aparecen y serán registrados en la base de datos del sistema como lo muestra la figura 4.2.2.4.

El usuario administrador, en el módulo de Consulta de Queja Pendientes, como su nombre lo indica, únicamente puede consultar las quejas pendientes que se encuentran hasta ese momento de resolver. (Vea figura 4.2.2.5).

En la figura 4.2.2.6 se muestra que el usuario puede realizar únicamente consultas, de las quejas atendidas, por un rango de fecha, por turno, por tipo de falla, por solución y/o por línea.

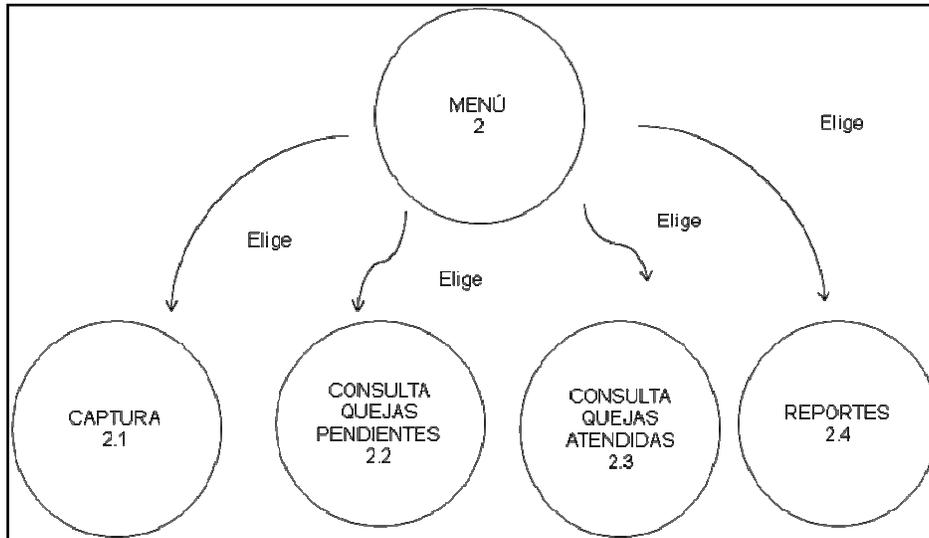


Figura 4.2.2.3 Diagrama de procesos de nivel 2 del Menú del Sistema.

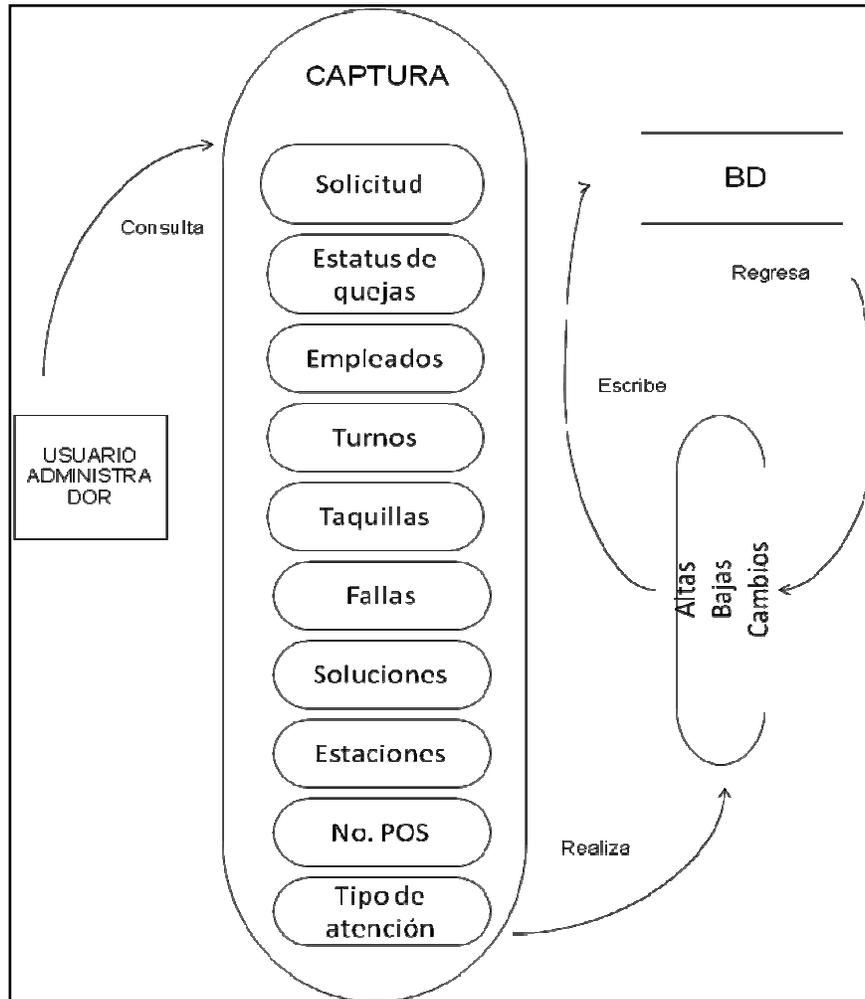


Figura 4.2.2.4 Diagrama de procesos de nivel 3 de Captura del Sistema.

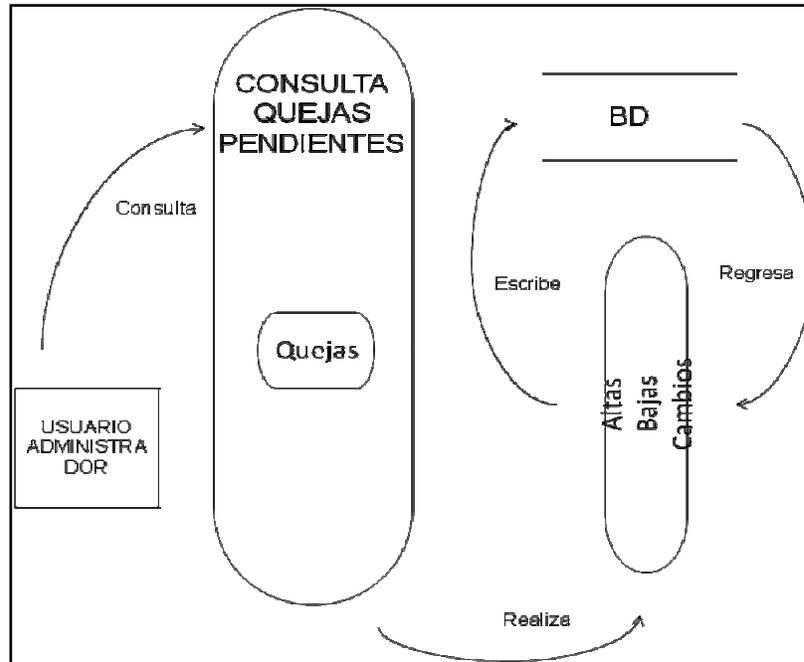


Figura 4.2.2.5 Diagrama de procesos de nivel 3 de Consulta del Sistema.

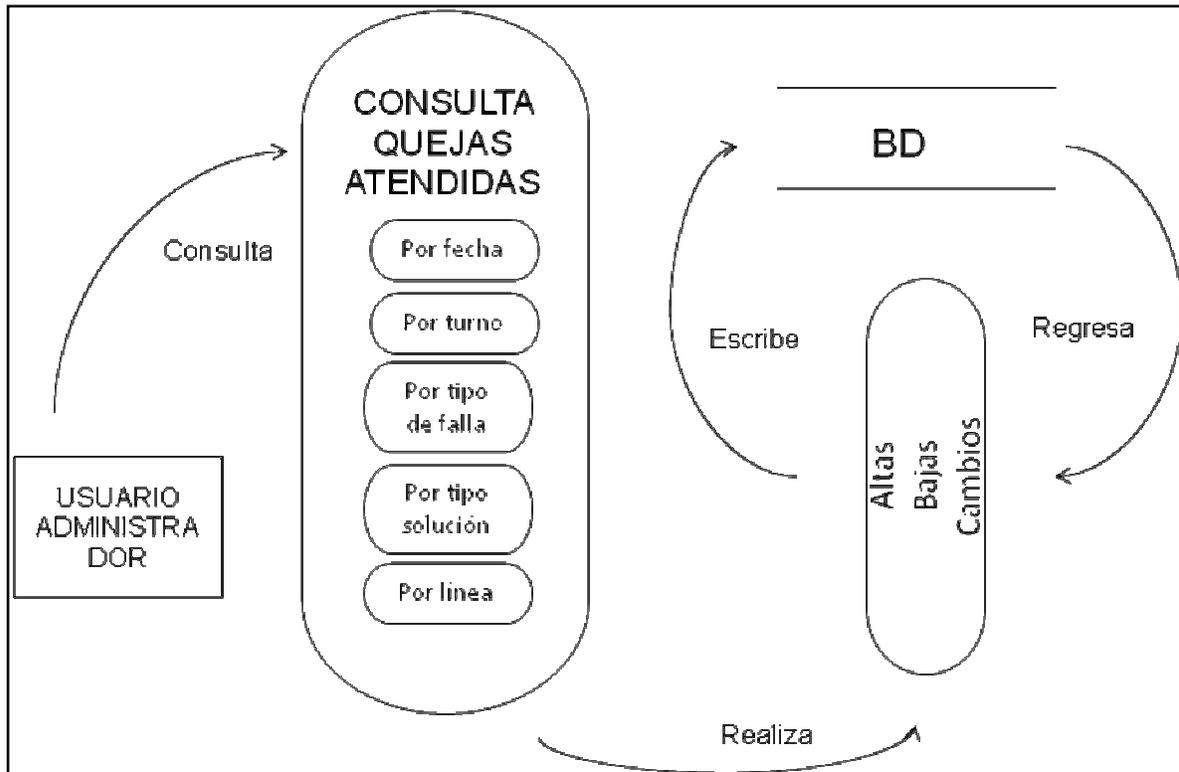


Figura 4.2.2.6 Diagrama de procesos de nivel 3 de Consulta del Sistema.

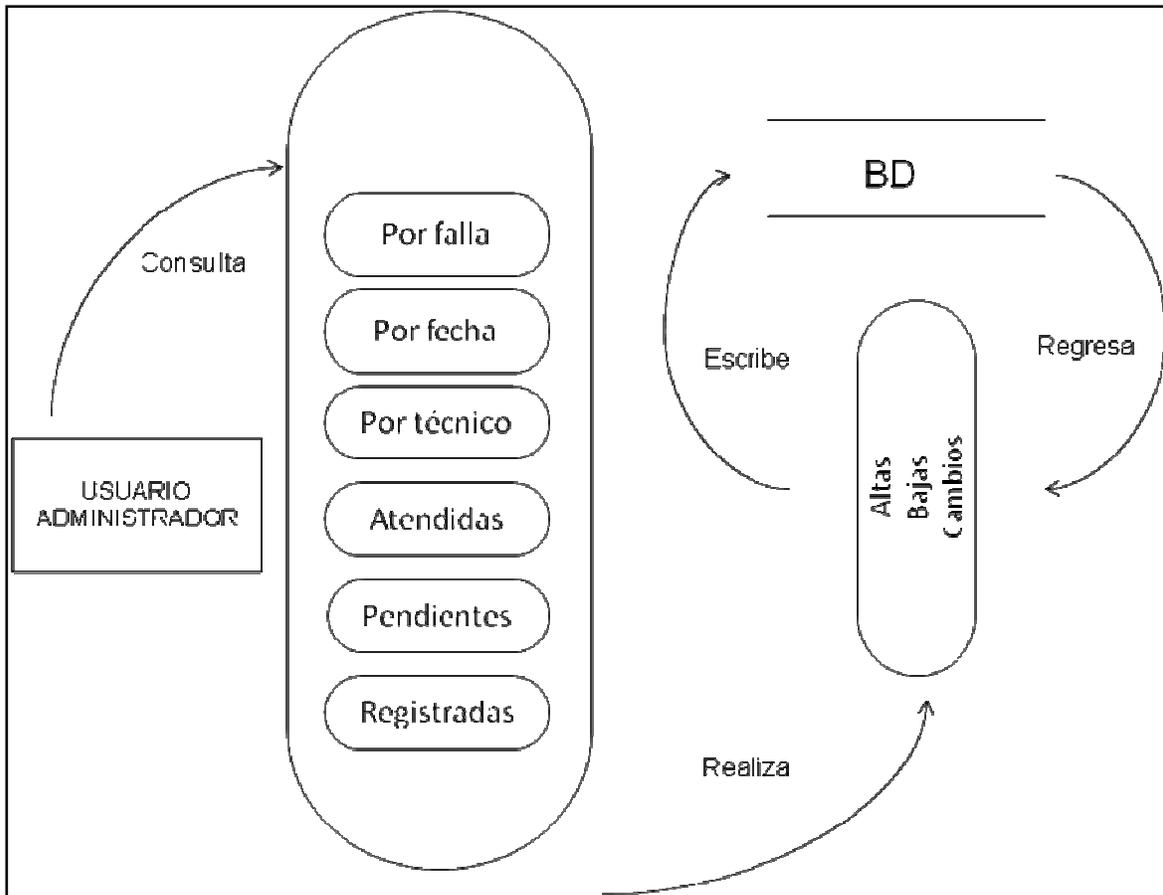


Figura 4.2.2.7 Diagrama de procesos de nivel 3 por Filtro del Sistema.

En este módulo el usuario administrador, puede generar diversos reportes de las opciones disponibles (Ver figura 4.2.2.7).

➤ Diagrama de bloques

En el diagrama de bloques (Vea figura 4.2.2.8) se puede observar de una forma general los 4 módulos que conforman a SIREPOS con sus opciones.

De igual forma en la figura 4.2.2.9 podemos observar el diagrama de flujo del sistema que se desarrollará en este trabajo de tesis.

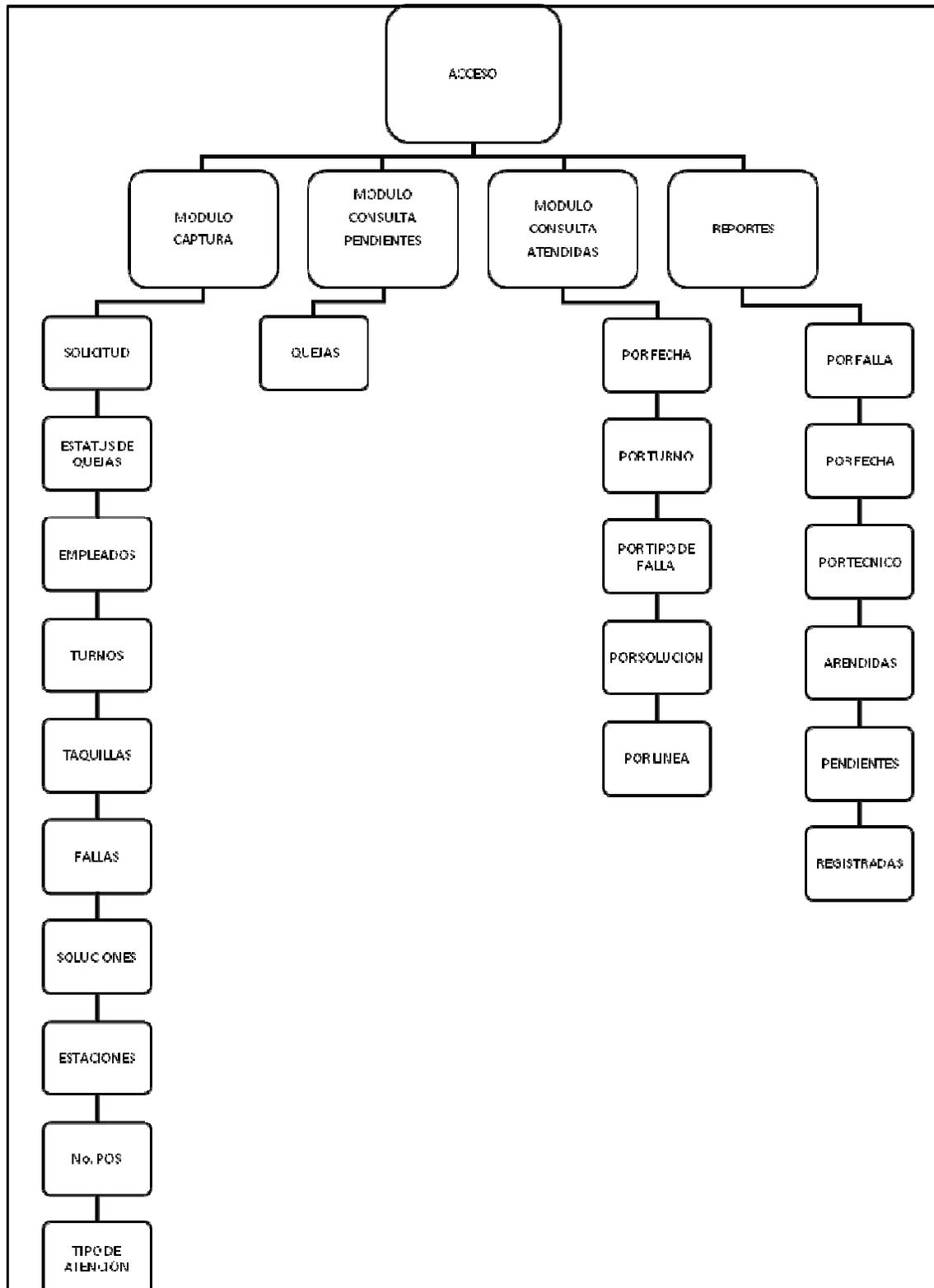


Figura 4.2.2.8 Diagrama de bloques del Sistema.

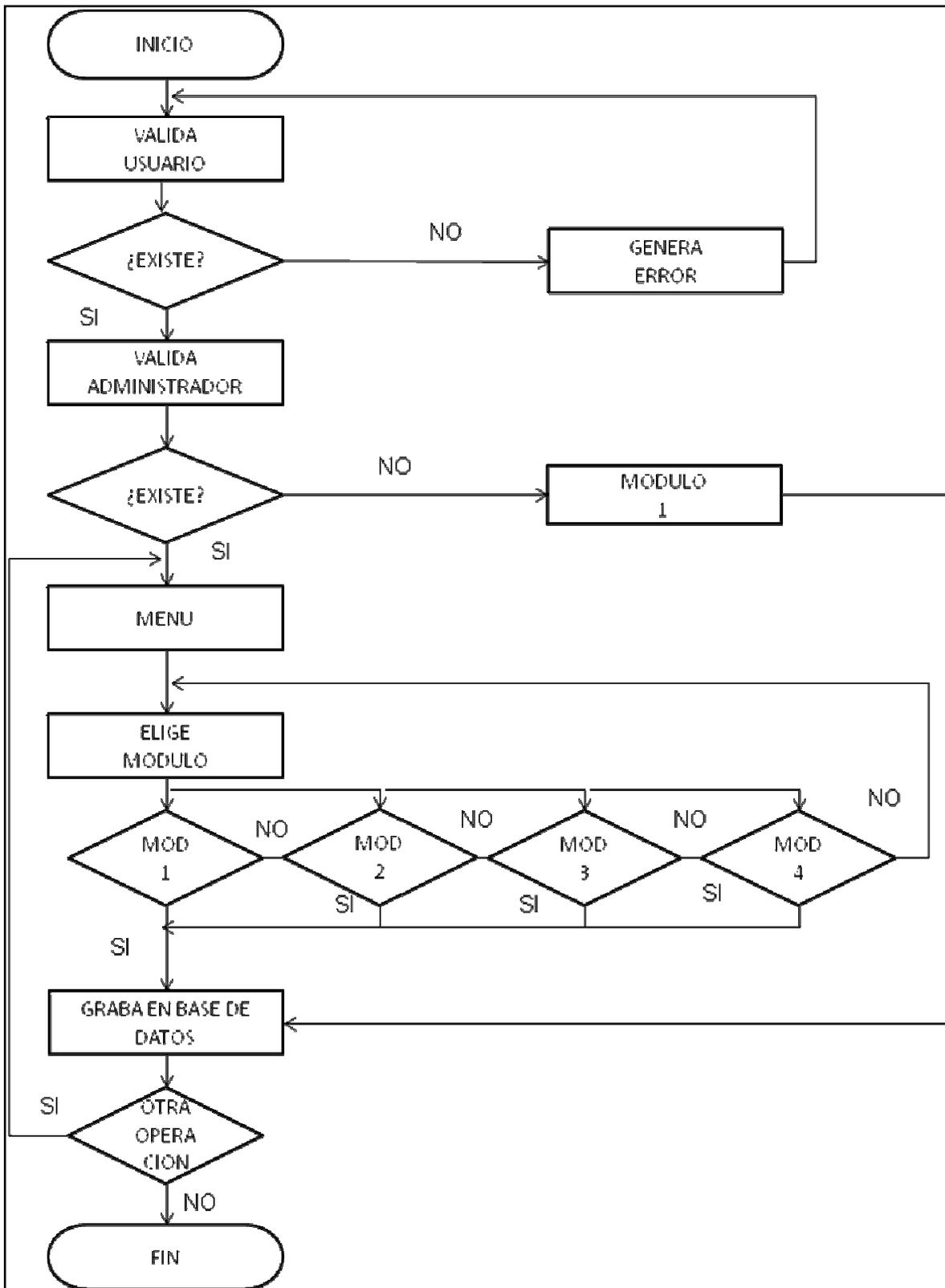


Figura 4.2.2.9 Diagrama de flujo del Sistema.



4.2.3 Diccionario de Datos.

El diccionario de datos es un catálogo que contiene los elementos de un sistema. Como su nombre lo sugiere, estos elementos se centran alrededor de los datos y la forma en que están estructurados para satisfacer los requerimientos de los usuarios y las necesidades de la organización. Los elementos más importantes que contiene este son:

- Las características lógicas de los sitios donde se almacenan los datos del sistema, incluyendo nombre, descripción, alias, contenido y organización.
- Los procesos donde se emplean los datos y los sitios donde se necesita el acceso inmediato a la información.
- Datos acerca de los datos que comúnmente son llamados metadatos.

Entre las principales razones para que un diccionario de datos sea utilizado se encuentran:

- ✓ Para manejar los detalles en el sistema, ya que tienen enormes cantidades de datos, aun en los sistemas más chicos.
- ✓ Manejar todos los detalles ya que los sistemas sufren cambios continuos. Auxilia en el análisis y diseño del software.
- ✓ Los analistas mas organizados usan el diccionario de datos específicamente para el análisis y diseño de software.
- ✓ Proporcionan asistencia para asegurar significados comunes para los elementos y actividades del sistema, registrando detalles adicionales relacionados con el flujo de datos, de tal manera que todo pueda localizarse con rapidez.
- ✓ Documentar las características del sistema, incluyendo partes o componentes así como los aspectos que los distinguen, procesos y frecuencia de estos.
- ✓ Facilitar el análisis de los detalles con la finalidad de evaluar las características y determinar donde efectuar cambios.
- ✓ Determina si son necesarias nuevas características o si están en orden los cambios de cualquier tipo.



- ✓ Localizar errores y omisiones en el sistema, detecta dificultades, y presentarlas en un informe. Aún en los manuales, se revelan errores.

El contenido de un registro del diccionario, ver figura 4.2.3.1, es un conjunto de columnas que contienen:

- ✓ Nombre de dato.
- ✓ Acrónimo.
- ✓ Tipo de dato.
- ✓ Longitud.
- ✓ Llave.
- ✓ Aceptación de nulos.
- ✓ Descripción.

NOMBRE DE TABLA						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN

Tabla 4.2.3.1 Descripción del contenido del Diccionario de Datos.

A continuación se muestran los diccionarios de datos para cada una de las tablas que se utilizarán en este trabajo de tesis:

- ✓ PERTAQ, vea tabla 4.2.3.2.
- ✓ ESTACION , vea tabla 4.2.3.3.
- ✓ TAQUILLA, vea tabla 4.2.3.4.
- ✓ FALLA, vea tabla 4.2.3.5.
- ✓ SOLUCION, vea tabla 4.2.3.6.
- ✓ TURNO, vea tabla 4.2.3.7.



- ✓ POS, vea tabla 4.2.3.8.
- ✓ ASESOR, vea tabla 4.2.3.9.
- ✓ ATENCION, vea tabla 4.2.3.10.
- ✓ ESTATUS, vea tabla 4.2.3.11.
- ✓ USUARIOS, vea tabla 4.2.3.12.
- ✓ UBIPOS, vea tabla 4.2.3.13.
- ✓ SOLICITUD, vea tabla 4.2.3.14.

NOMBRE DE TABLA: PERTAQ						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Expediente Personal de Taquilla	exp_taq	Number	9	PK	No	Número de expediente del personal de taquilla
Nombre Personal de Taquilla	nom_taq	Char	255		Si	Contiene el nombre del personal de taquilla
Apellido Materno Personal de Taquilla	ape_mat_taq	Char	255		Si	Contiene el apellido materno del personal de taquilla
Apellido Paterno Personal de Taquilla	ape_pat_taq	Char	255		Si	Contiene el apellido paterno del personal de taquilla

Tabla 4.2.3.2 Descripción de PERTAQ.



NOMBRE DE TABLA: ESTACION						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Clave estación	cve_est	Number	9	PK	No	Número de estación
Descripción estación	dsc_est	Char	255		Si	Nombre de la estación
Línea	linea	Char	5		Si	Número de Línea

Tabla 4.2.3.3 Descripción de ESTACION.

NOMBRE DE TABLA: TAQUILLA						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Clave taquilla	cve_taq	Number	9	PK	No	Contiene el número de la taquilla
Descripción taquillas	dsc_taq	Char	18		Si	Descripción de la taquilla

Tabla 4.2.3.4 Descripción de TAQUILLA.

NOMBRE DE TABLA: FALLA						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Clave falla	cve_falla	Number	9	PK	No	Identifica el tipo de falla
Descripción falla	dsc_falla	Char	255		Si	Tipo de falla

Tabla 4.2.3.5 Descripción de FALLA.



NOMBRE DE TABLA: SOLUCION						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Clave solución	cve_sol	Number	9	PK	No	Tipo de solución
Descripción solución	dsc_sol	Char	255		Si	Indica el tipo de solución

Tabla 4.2.3.6 Descripción de SOLUCION.

NOMBRE DE TABLA: TURNO						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Clave turno	cve_tur	Number	9	PK	No	Número de Turno
Descripción	dsc_tur	Char	30		Si	Indica el turno del personal

Tabla 4.2.3.7 Descripción de TURNO.

NOMBRE DE TABLA: POS						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Clave POS	cve_pos	Number	9	PK	No	Indica el número de POS
POS	dsc_pos	Number	9		Si	Descripción del POS

Tabla 4.2.3. 8 Descripción de POS.



NOMBRE DE TABLA: ASESOR						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Expediente asesor	exp_ase	Number	9	PK	No	Expediente del personal de soporte técnico
Nombre asesor	nom_ase	Char	255		Si	Nombre del personal de soporte técnico
Apellido Paterno asesor	ape_pat_ase	Char	255		Si	Apellido paterno del personal de soporte técnico
Apellido Materno asesor	ape_mat_ase	Char	255		Si	Apellido materno del personal de soporte técnico

Tabla 4.2.3.9 Descripción de ASESOR.

NOMBRE DE TABLA: ATENCION						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Clave atención	cve_atn	Number	9	PK	No	Identificador de la atención
Atención	dsc_atn	Char	255		Si	Descripción del tipo de atención

Tabla 4.2.3.10 Descripción de ATENCION.



NOMBRE DE TABLA: ESTATUS						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Clave estatus	cve_estatus	Number	9	PK	No	Identificador del estatus
Estatus	dsc_estatus	Char	10		Si	Descripción del estatus de una falla

Tabla 4.2.3.11 Descripción de ESTATUS.

NOMBRE DE TABLA: USUARIOS						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Clave usuario	cve_usr	Char	18	PK	No	Nombre de usuario
Password	pwd	Char	15		No	Contraseña de usuario
Roll	roll	Char	15		No	Identifica el tipo de usuario

Tabla 4.2.3.12 Descripción de USUARIOS.

NOMBRE DE TABLA: UBIPOS						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Clave POS	cve_pos	Number	9	PK	No	Indica el número de POS
Clave estación	cve_est	Number	9	PK	No	Número de estación

Tabla 4.2.3.13 Descripción de UBIPOS.



NOMBRE DE TABLA: SOLICITUD						
NOMBRE DE DATO	ACRÓNIMO	TIPO DE DATO	LONGITUD	LLAVE	NULOS	DESCRIPCIÓN
Solicitud	no_solicitud	Number	9	PK	No	Número de solicitud
Clave POS	cve_pos	Number	9	PK	No	Indica el número de POS
Clave falla	cve_falla	Number	9	PK	No	Identifica el tipo de falla
Clave solución	cve_sol	Number	9	PK	No	Tipo de solución
Clave turno	cve_tur	Number	9	PK	No	Número de Turno
Expediente asesor	exp_ase	Number	9	PK	No	Expediente del personal de soporte técnico
Clave atención	cve_atn	Number	9	PK	No	Identificador de la atención
Clave estatus	cve_estatus	Number	9	PK	No	Identificador del estatus
Expediente Personal de Taquilla	exp_taq	Number	9	PK	No	Número de expediente del personal de taquilla
Fecha Alta	fecha_alta	datetime	8		No	Fecha en la que se genera una solicitud
Hora Alta	hora_alta	datetime	8		No	Hora en la que se genera una solicitud
Fecha atención	Fecha_atn	datetime	8		No	Fecha en la que queda resulta una solicitud
Hora atención	hora_atn	datetime	8		No	Hora en la que se resuelve una solicitud

Tabla 4.2.3.13 Descripción de SOLICITUD.



4.2.4 Diagrama Entidad Relación.

Los diagramas entidad-relación (**a veces denominado por su siglas, E-R “Entity relationship”**) son una herramienta para el modelado de datos de un sistema de información. Estos modelos expresan entidades relevantes para un sistema de información, sus inter-relaciones y propiedades.

Los diagramas E-R son un lenguaje gráfico para describir conceptos. Informalmente, son simples dibujos o gráficos que describen la información que trata un sistema de información y el software que lo automatiza. Los elementos de dicho lenguaje se describen a continuación, por orden de importancia:

- **ENTIDADES:** Una entidad es cualquier “objeto” discreto sobre el que se tiene información.
- **RELACIONES:** Una relación describe cierta interdependencia (de cualquier tipo) entre entidades. Una relación no tiene sentido sin las entidades que relaciona.
- **ATRIBUTOS:** Los atributos son propiedades relevantes propias de una entidad y/o relación.

Cardinalidad de las Relaciones.

Las relaciones, en principio binarias, pueden involucrar a un número distinto de instancias de cada entidad. Así, son posibles tres tipos de cardinalidades:

- **Relaciones de uno a uno:** una instancia de la entidad A se relaciona con una y solamente una de la entidad B.
- **Relaciones de uno a muchos:** cada instancia de la entidad A se relaciona con varias instancias de la entidad B.
- **Relaciones de muchos a muchos:** cualquier instancia de la entidad A se relaciona con cualquier instancia de la entidad B.



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

La figura 4.2.4.1 Muestra el Diagrama Entidad Relación de nuestro Sistema.

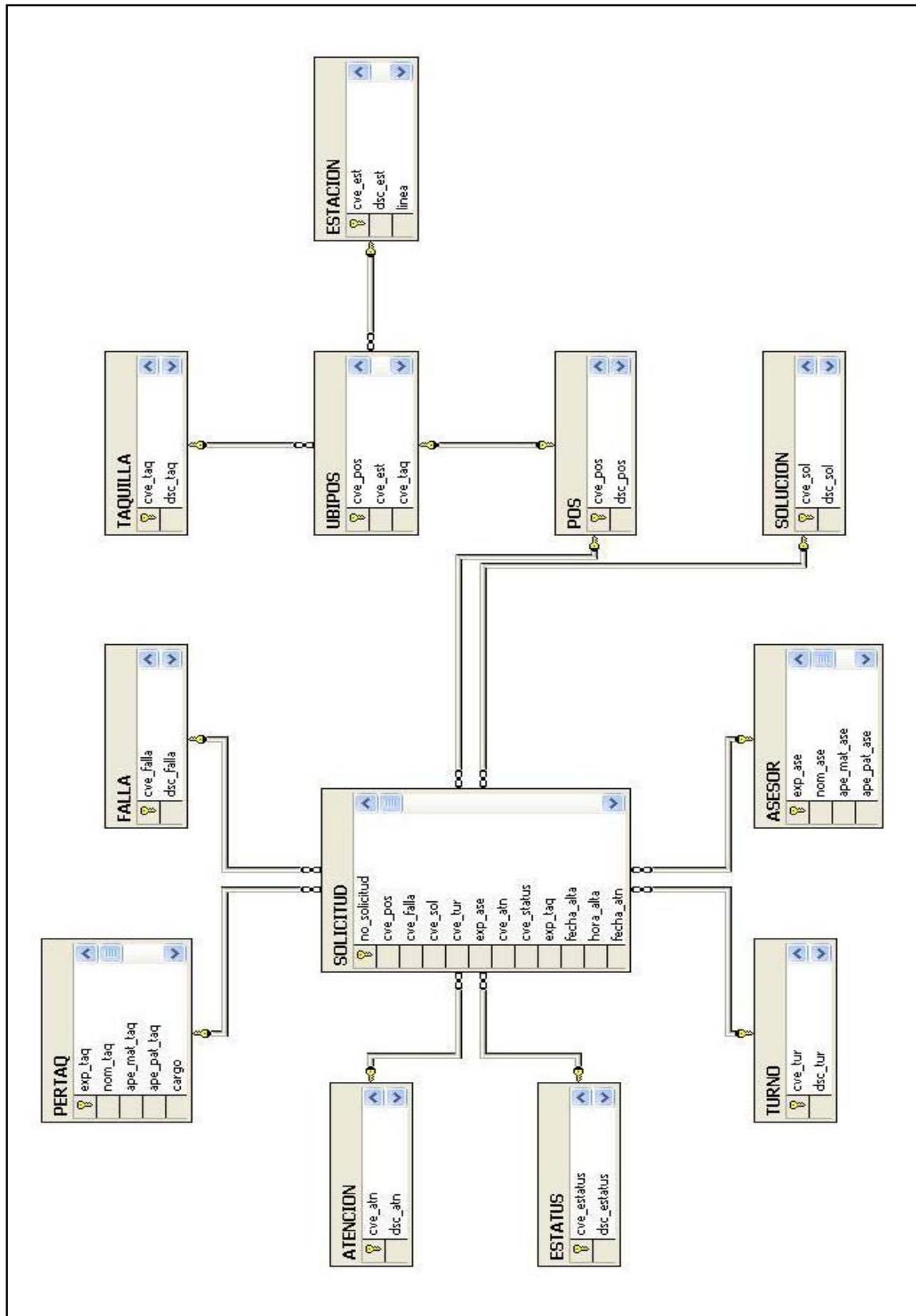


Figura 4.2.4.1 Diagrama Entidad Relación.



4.2.5 Normalización.

En este apartado del presente trabajo se aplicará de forma práctica el concepto de Normalización que como ya se mencionó en el capítulo 2 se trata de aplicar una serie de pasos que nos permitirán tener un conjunto de datos bien organizados, no repetitivos y sobre todo congruentes entre ellos.

Los datos con los que se trabaja en el área de Soporte Técnico al POS para la captura de reportes son: Numero de solicitud, Expediente de la supervisora que reporta, Nombre de la supervisora que reporta, Número de taquilla, Línea, Estación, Numero de POS que presenta la falla, Turno de Taquilla en que se presenta la falla, Fecha de alta de la falla, Hora de alta de la falla, Tipo de falla, Tipo de solución, Expediente de quien recibe la llamada de reporte de falla, Nombre de quien recibe la falla, Expediente de quien atiende la falla, Nombre de quien atiende la falla, Turno de quien atiende la falla, Tipo de atención, Fecha de atención, Hora de atención, Expediente del personal de taquilla, Nombre del personal de taquilla, Turno de atención en taquilla y Estatus de la falla.

a) Primera Forma Normal.

Como primer paso, es necesario asignar identificadores para cada uno de los campos en vez de los nombres completos tal y como se utilizan ahora.

Tomando en cuenta lo que la Primera forma Normal, nos dice: “cada atributo debe tener un valor atómico para cada uno de sus elementos”, en caso de que esta regla no se cumpla, es necesario separar los atributos que tengan mas de un valor en la intersección fila – columna formando nuevos campos.

La siguiente tabla (figura 4.2.5.1) representa la primera forma normal:



Solicitud	
PK	No_solicitud
	Nom_taq
	Ape_mat_taq
	Ape_pat_taq
	Estación
	Línea
	POS
	Turno
	Fecha_alta
	Hora_alta
	Estación
	Falla
	Solución
	Nombre_ase
	Ape_mat_ase
	Ape_pat_ase
	Atención
	Fecha_atn
	Hora_atn
	Estatus

Figura 4.2.5.1 Tabla en 1FN.

b) Segunda Forma Normal.

Ahora para lograr que nuestra tabla cumpla con la Segunda Forma Normal que nos dice: “todos los atributos de la relación dependen del identificador único”, es necesario identificar los grupos de datos que pueden ser separados debido a su interdependencia y generar nuevos grupos o tablas. Esto con el objetivo de que: Cada instancia específica del identificador único determine solo a una instancia simple de cada atributo. En caso de que un atributo dependa de una parte del



identificador único, entonces está mal ubicado y este atributo debe de ser movido a otro grupo.

Para nuestro caso concreto se observan varias interdependencias que podrán ser reducidas a un solo grupo por que dependen de un solo atributo.

Por ejemplo los datos del personal de taquilla pueden ser agrupados en una sola tabla, llamada PERTAQ en donde cada uno de los campos dependerá de un solo campo llamado exp_taq, como lo muestra la tabla 4.2.5.2.

PERTAQ	
PK	exp_taq
	nom_taq
	ape_mat_taq
	ape_pat_taq
	cargo

Figura 4.2.5.2 Tabla PERTAQ en 2FN.

Así mismo aplicando la segunda forma normal al resto de los campos de la tabla 4.2.5.1, nos encontramos que pueden ser reducidos a las tablas 4.2.5.3, 4.2.5.4, 4.2.5.5, 4.2.5.6, 4.2.5.7, 4.2.5.8, 4.2.5.9, 4.2.5.10, 4.2.5.11 las cuales se muestran a continuación.

ESTACION	
PK	cve_est
	dsc_est
	linea

Figura 4.2.5.3 Tabla ESTACION en 2FN.



TAQUILLA	
PK	cve_taq
	dsc_taq

Figura 4.2.5.4 Tabla TAQUILLA en 2FN.

FALLA	
PK	cve_falla
	dsc_falla

Figura 4.2.5.5 Tabla FALLA en 2FN.

SOLUCION	
PK	cve_sol
	dsc_sol

Figura 4.2.5.6 Tabla SOLUCION en 2FN.

TURNO	
PK	cve_tur
	dsc_tur

Figura 4.2.5.7 Tabla TURNO en 2FN.

POS	
PK	cve_pos
	dsc_pos

Figura 4.2.5.8 Tabla POS en 2FN.



ASESOR	
PK	exp_ase
	nom_ase
	ape_mat_ase
	ape_pat_ase

Figura 4.2.5.9 Tabla ASESOR en 2FN.

ATENCIÓN	
PK	cve_atn
	dsc_atn

Figura 4.2.5.10 Tabla ATENCIÓN en 2FN.

ESTATUS	
PK	cve_estatus
	dsc_estatus

Figura 4.2.5.11 Tabla ESTATUS en 2FN.

Cuando se alcanza el nivel de la Segunda Forma Normal, se controlan la mayoría de los problemas de lógica y se puede insertar un registro sin exceso de datos en la mayoría de las tablas.

Es necesario incorporar una tabla más, (Vea figura 4.2.5.12) para el control de los usuarios que pueden acceder a la base por medio de una clave de usuario y un password y el perfil que tienen (usuario o administrador).



USUARIOS	
PK	cve_usr
	pwd
	roll

Figura 4.2.5.12 Tabla de USUARIOS en 2FN.

c) Tercera forma normal.

La Tercera Forma Normal nos dice que: Una tabla además de cumplir con la 2FN, deben eliminarse las dependencias transitivas, es decir, ningún atributo considerado no principal depende transitivamente de otro atributo que tampoco esta considerado como llave primaria y que en caso de presentarse esta situación se deben mover los atributos dependientes y el atributo del cual dependen a una nueva tabla, nombrar el atributo del cual dependen como identificador único y repetirlo en el grupo original para mantener la relación.

Si las tablas están en la Tercera Forma Normal se previenen errores de lógica debido a que cada columna en una tabla está identificada de manera única por la llave primaria.

En general, la regla de la Tercera Forma Normal señala que hay que eliminar y separar cualquier dato que no sea clave. El valor de esta columna debe depender de la clave. Todos los valores deben identificarse únicamente por la clave.

Por lo anterior, para obtener la ubicación precisa de un POS es necesario crear una tercera tabla, aplicando la tercera forma normal la cual llamaremos UBIPOS la cual se muestra en la tabla 4.2.5.13.



UBIPOS	
PK	cve_pos
FK	cve_est
FK	cve_taq

Figura 4.2.5.13 Tabla de UBIPOS en 3FN.

Por último después de aplicar la segunda y tercera forma normal a la tabla 4.2.5.1, tendremos la tabla 4.2.5.14.

SOLICITUD	
PK	no_solicitud
FK	cve_pos
FK	cve_falla
FK	cve_sol
FK	cve_tur
FK	exp_ase
FK	cve_atn
FK	cve_status
FK	exp_taq
	fecha_alta
	hora_alta
	fecha_atn
	hora_atn

Figura 4.2.5.14 Tabla de SOLICITUD en 3FN.

4.3 Diseño y construcción del back-end.

El back-end es la parte de la aplicación que nos permitirá determinar cómo se almacenará la información, tiene por objetivo crear la estructura del sistema modular, para facilitar el proceso de captura de datos, el proceso de búsqueda de



información, la actualización y manipulación de los mismos. Todo esto será transparente para el usuario final ya que los maneja a través del front-end del cual hablaremos más adelante.

Las bases de datos tienen como funcionalidad el almacenar la información evitando la redundancia de datos y permitiendo que ésta se pueda recuperar de forma rápida, segura, que sean verídicos y que al ser modificados no se afecte la integridad de la base.

Por lo que en este subcapítulo crearemos una base de datos, tablas, consultas, procedimientos almacenados, triggers de forma gráfica así como con sentencias de SQL, así mismo generaremos el diagrama entidad-relación.

➤ **Creación de una base de datos.**

Para crear una base de datos en SQL Server debe de estar formada por un conjunto de tablas en las que se almacenan datos estructurados. Las tablas contienen diversos tipos de controles que garantizan la validez de los datos y pueden presentar índices, similares a los de los libros, que permiten localizar las filas rápidamente. La creación de la base puede hacerse de dos maneras a través del Administrador Corporativo o mediante sentencias SQL.

Para crear la base de datos que nos servirá para crear nuestras tablas de nuestro Sistema ingresamos al SQL Administrador Corporativo como se muestra en la figura 4.3.1.

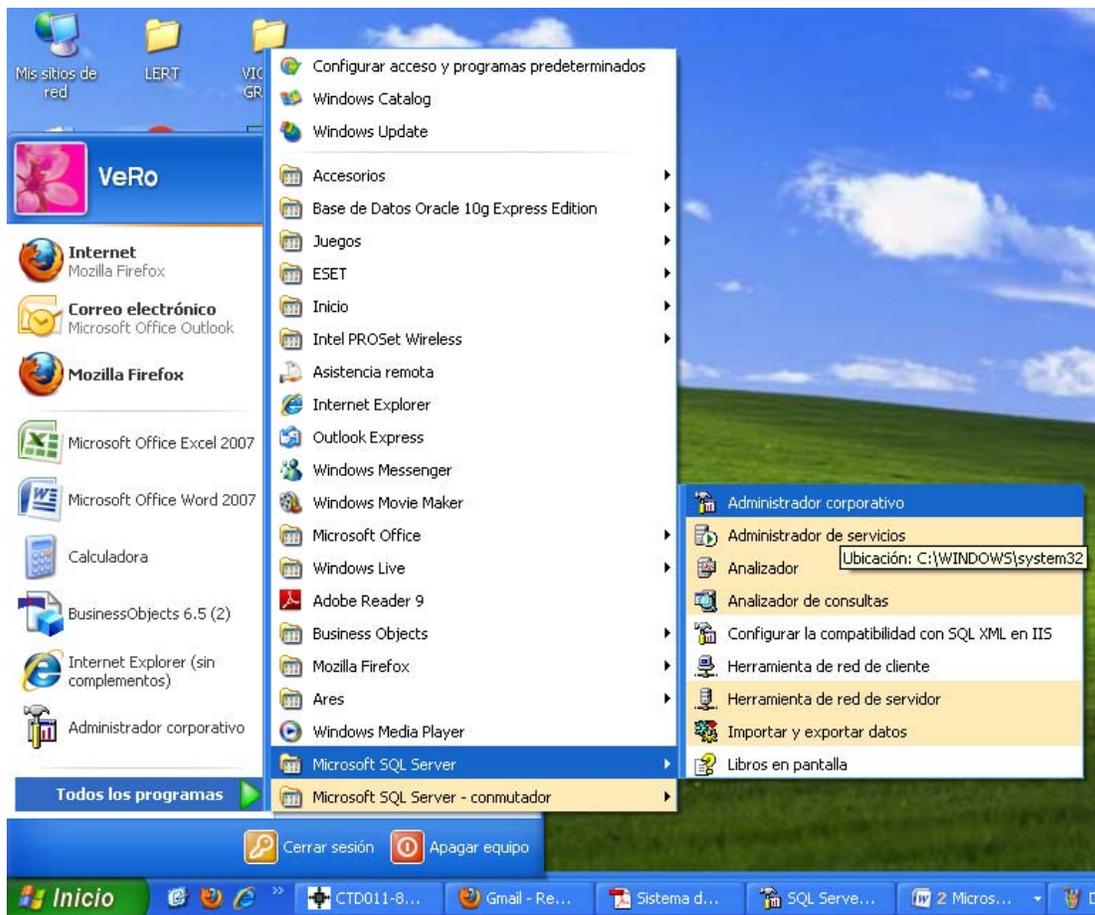


Figura 4.3.1 Ingreso al Administrador Corporativo de SQL.

Posteriormente aparecerá una pantalla que nos pide ingresar ciertos datos como: Servidor y los datos de usuario para la conexión en caso de querer realizar la conexión hacia otro servidor. Los cuales deberán ser llenados con los datos correspondientes al servidor, o bien, la computadora donde estará la Base de Datos que será utilizada.

En nuestro caso en Server utilizaremos local que hace referencia a nuestra computadora, así como la autenticación de Windows ya que en este momento no tenemos ningún usuario creado. En caso de tener alguno con permisos de administrador se puede cambiar el método de autenticación. (Vea figura 4.3.2).



Figura 4.3.2 Propiedades de SQL.

Para crear nuestra nueva base de datos que vamos a utilizar mediante el Administrador Corporativo, debemos situarnos en Servidores de Microsoft SQL Server\Grupo de SQL Server\Local\Bases de Datos y ahí dar clic derecho y posteriormente en Nueva base de datos como lo muestra la figura 4.3.3.

En el proceso de creación de una base de datos se puede definir su nombre, establecer sus propiedades y fijar la ubicación de sus archivos.

La ficha Archivos de base de datos especifica el nombre, ubicación, espacio asignado y grupo de archivos, de los archivos de datos seleccionados (en el campo Archivo de transacciones). Además presenta los siguientes campos: Eliminar, crecimiento automático del archivo (en megabytes o en porcentaje) y Tamaño máximo del archivo. Ver figura 4.3.3.

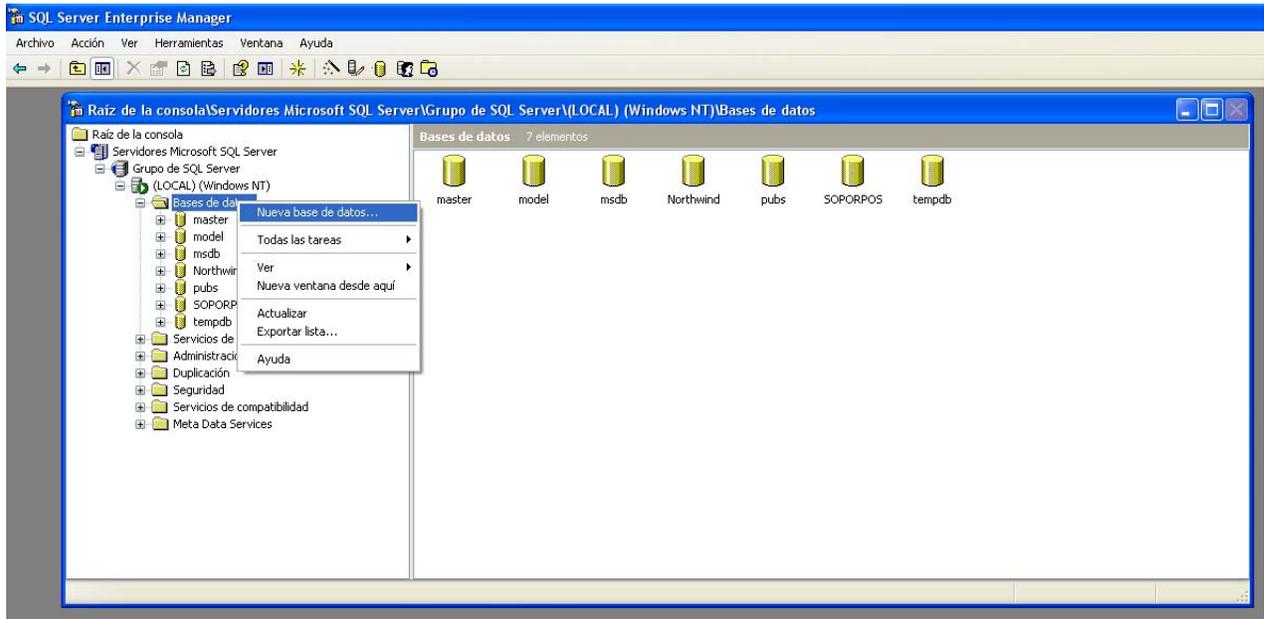


Figura 4.3.3 Creación de una nueva base de datos mediante el Administrador Corporativo de SQL Server.

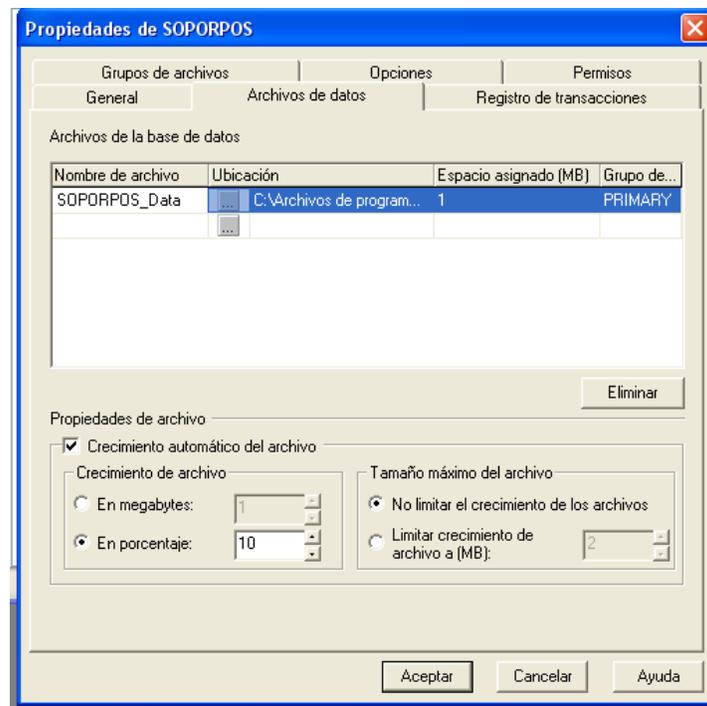


Figura 4.3.4 Propiedades de la Base de datos.



Para crear una Base de datos mediante sentencias SQL debemos situarnos en Servidores de Microsoft SQL Server\Grupo de SQL Server\Local\Bases de datos dar clic en el menú Herramientas y ahí dar clic en la opción Analizador de consultas SQL, y se abrirá una ventana en la cual podremos introducir el query correspondiente. En esta ocasión la sentencia a utilizar es create database, y el nombre de la tabla a crear; la cual llamaremos SOPORTE. (Vea figura 4.3.4).

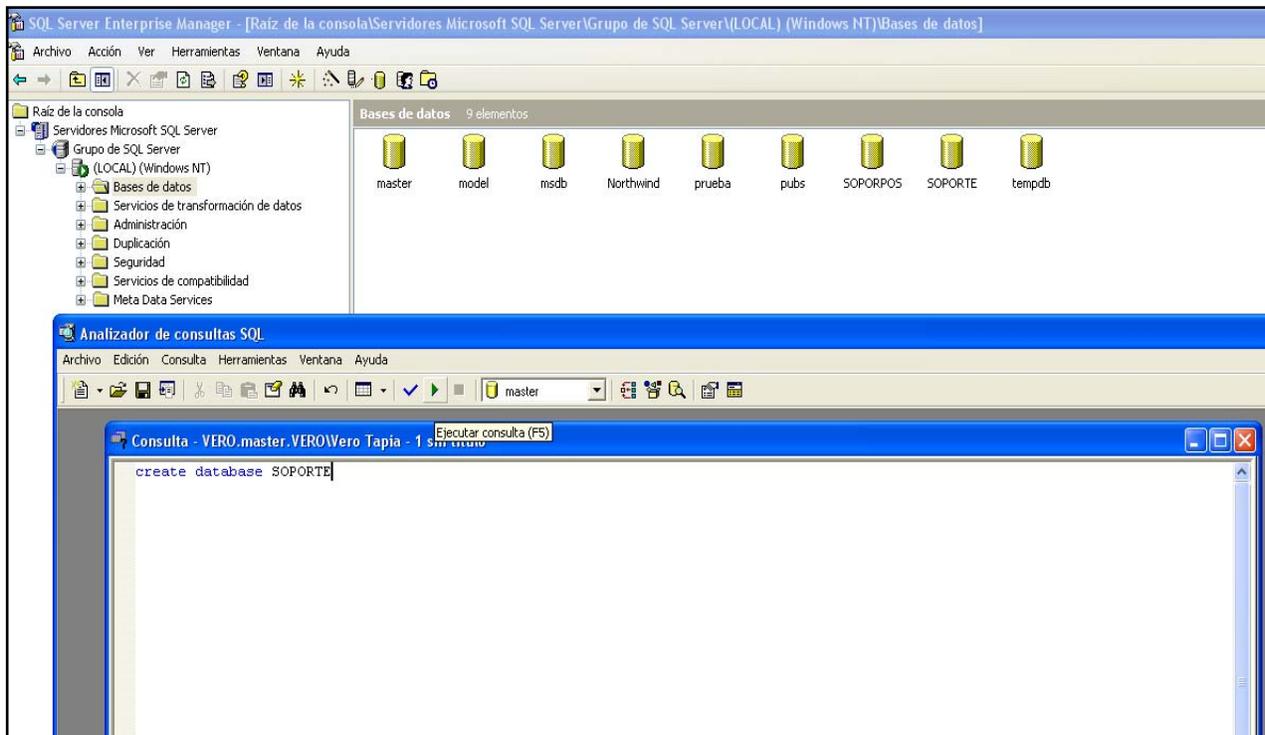


Figura 4.3.4 Creación de la base mediante secuencia SQL.

➤ **Creación de una tabla**

Una vez que se ha creado la base de datos, debemos dar clic derecho sobre el nombre de la misma, después nuevo y posteriormente tabla; y se abrirá una ventana, en donde podremos introducir cada una de los campos que contendrá dicha tabla, así como el tipo de dato, longitud, aceptación de nulos y definición de llaves, una vez que se han creado todos los campos se deberá guardar la tabla, y en ese momento se abrirá una pantalla donde deberá poner el nombre de la tabla, para este caso crearemos la tabla ATENCION (Vea figura 4.3.5).

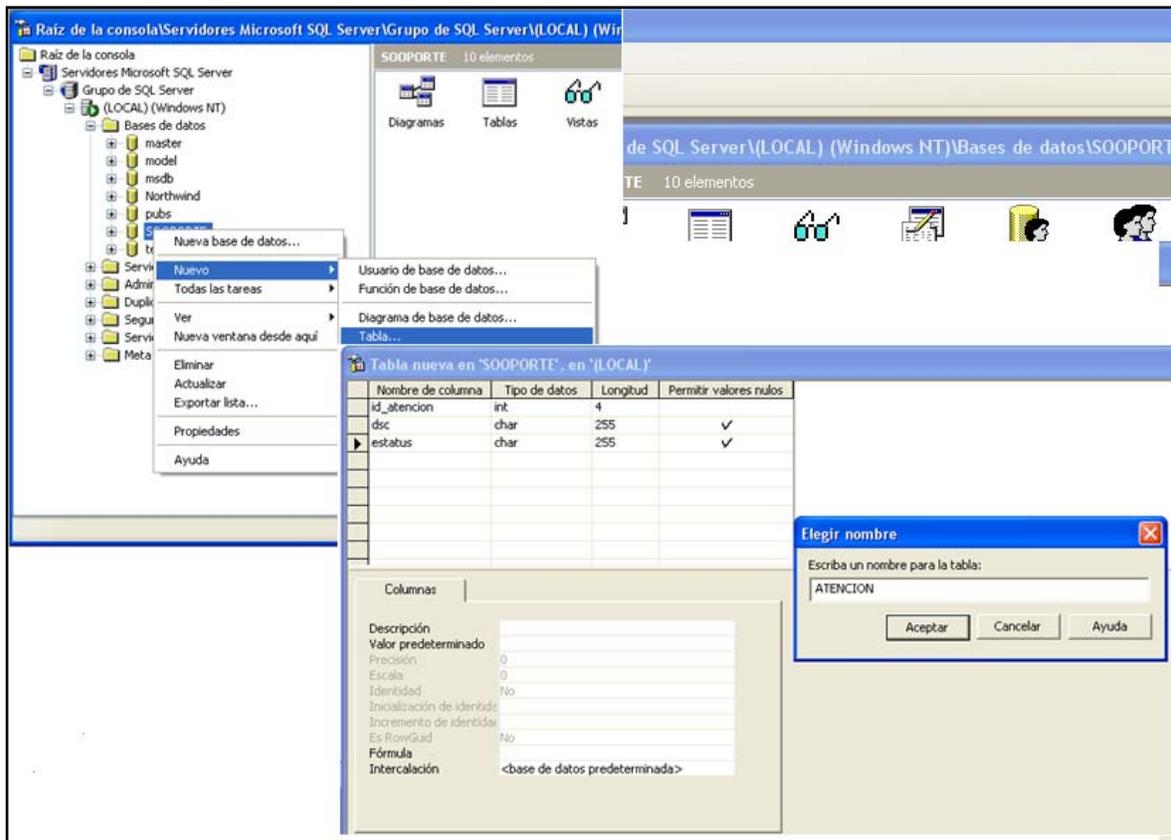


Figura 4.3.5 Creación de una nueva tabla mediante Administrador Corporativo.

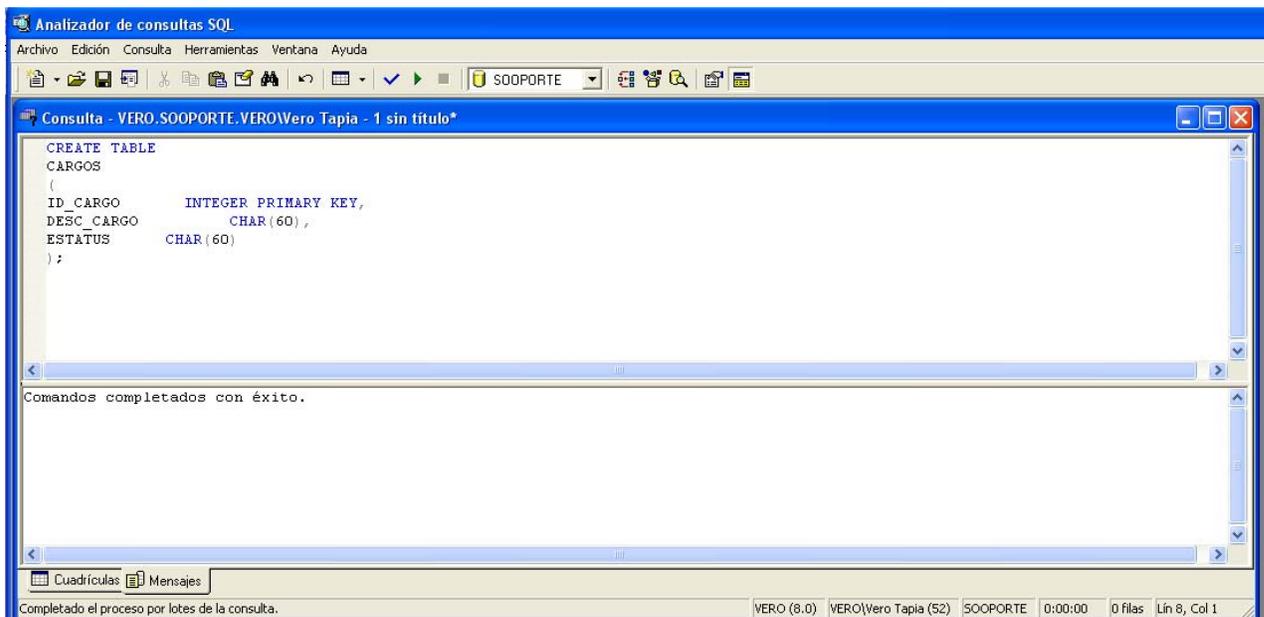


Figura 4.3.6 Creación de una nueva tabla mediante sentencias SQL.



Para crear una tabla con sentencias SQL, debemos situarnos en Servidores de Microsoft SQL Server\Grupo de SQL Server\Local\Bases de datos \SOPORTE dar clic en el menú Herramientas y ahí dar clic en la opción Analizador de consultas SQL, y se abrirá una ventana en la cual podremos introducir el query correspondiente. En esta ocasión la sentencia a utilizar es create table, además debemos introducir el nombre de la tabla, así como los campos que la conformarán, tipo, longitud y llaves. (Vea figura 4.3.6).

La Figura 4.3.7 muestra el árbol de las tablas creadas para nuestro sistema.

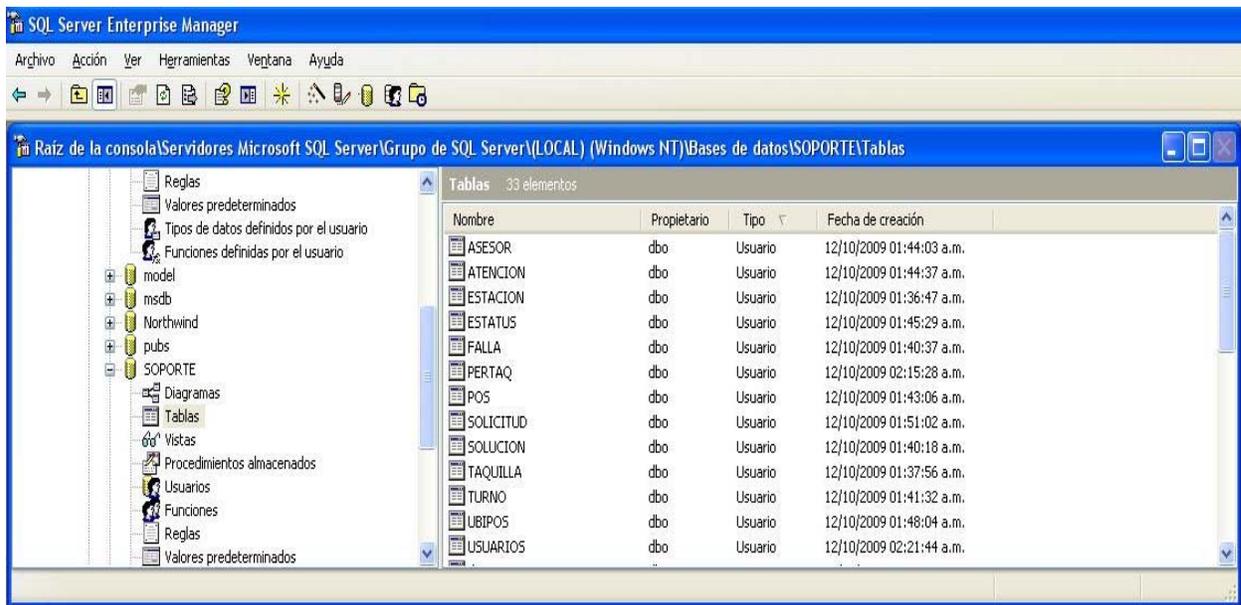


Figura 4.3.7 Árbol tablas creadas.

➤ Creación de una consulta.

Para crear una consulta con el Administrador Corporativo se debe dar clic derecho sobre la tabla en la que se desea hacer la consulta, Abrir tabla/consulta y aparecerá la ventana en la que podremos realizar la consulta, basta con elegir los campos que deseamos aparezcan en la consulta deseada (Vea figura 4.3.8).

Dentro de los principales iconos que tenemos en la pantalla de consultas, se encuentran los siguientes:



- Mostrar u ocultar el panel de diagrama. Nos permite visualizar o no las tablas que se están utilizando para realizar la consulta.
- Mostrar u ocultar el panel de cuadrícula. Nos permite visualizar o no en forma de tabla los campos que se eligen para realizar la consulta.
- Mostrar u ocultar el panel de SQL. Nos permite visualizar o no el query que se va generando para la consulta.
- Mostrar u ocultar el panel de resultados. Nos permite visualizar o no los resultados de la consulta.
- Ejecutar. Ejecuta una consulta.
- Comprobar SQL. Nos permite comprobar el código SQL para una consulta, antes de ejecutarla.
- Utilizar Group By. Nos permite agrupar las consultas por lo que el usuario desee.
- Agregar tabla. Nos permite agregar una tabla para consultar.

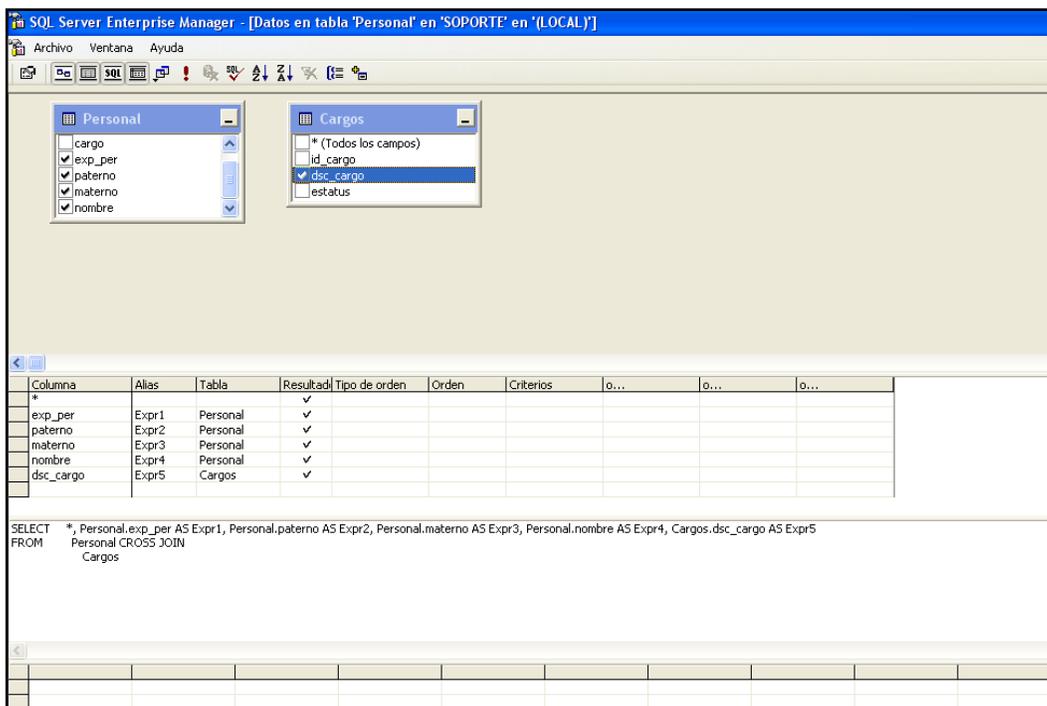


Figura 4.3.8. Creación de consulta por Administrador Corporativo.

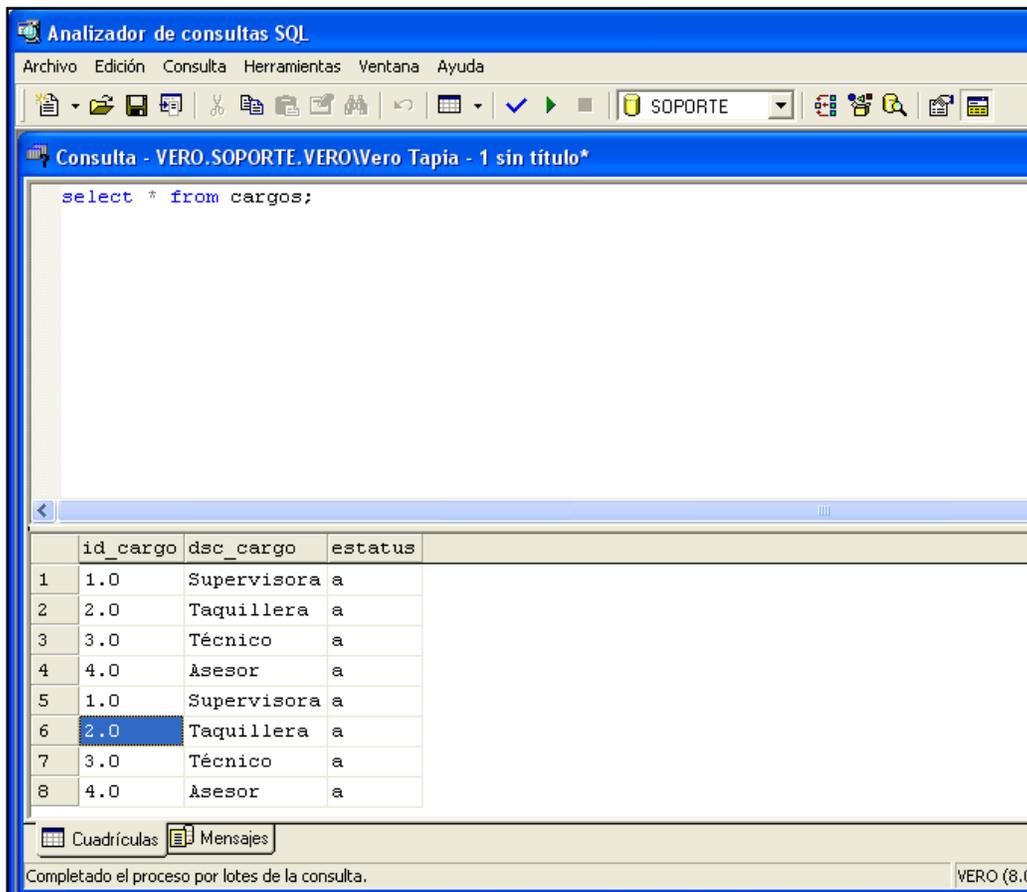


Figura 4.3.9. Creación de una consulta mediante sentencias SQL.

Para crear una consulta con sentencias SQL, debemos situarnos en Servidores de Microsoft SQL Server\Grupo de SQL Server\Local\Bases de datos \SOPORTE dar clic en el menú Herramientas y ahí dar clic en la opción Analizador de consultas SQL, y se abrirá una ventana en la cual podremos introducir el query correspondiente. En esta ocasión realizaremos una consulta donde nos muestre todo lo que contiene la tabla cargos. (Vea figura 4.3.9).

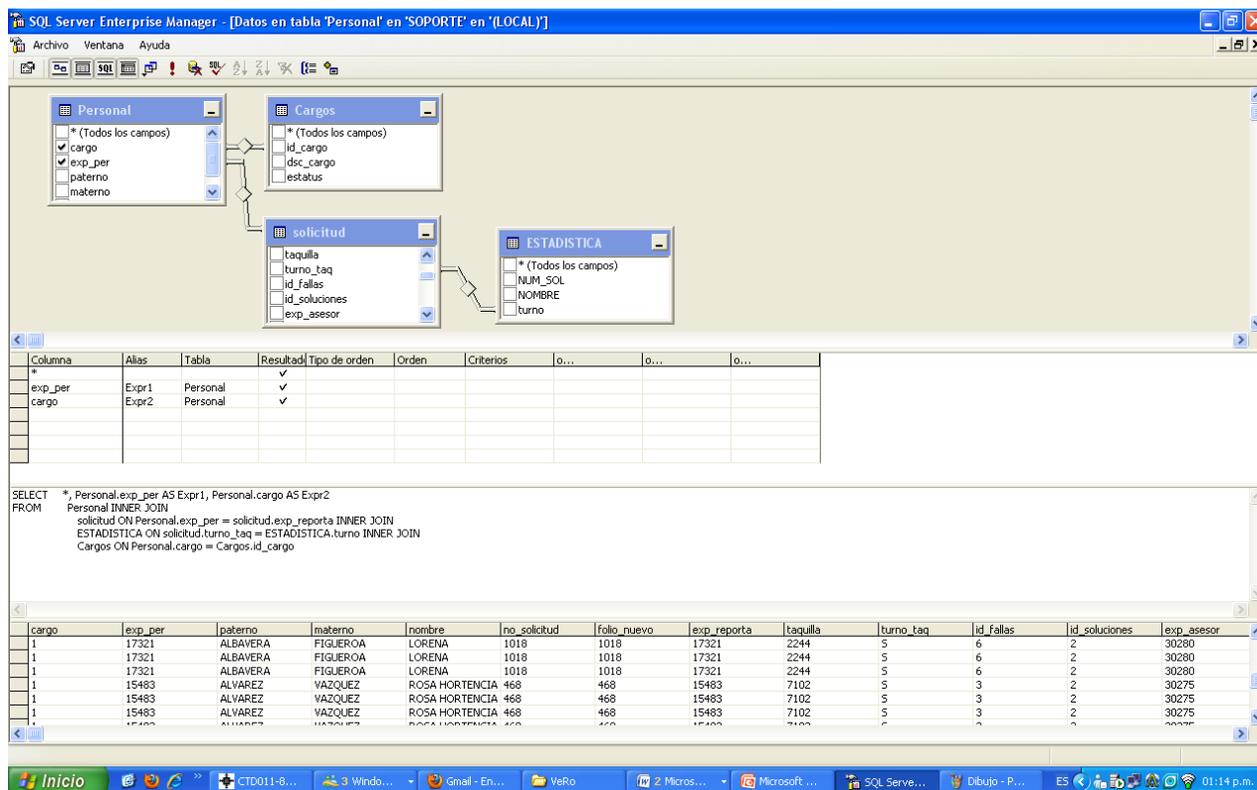


Figura 4.3.10 Creación de los Join.

SQL Server también nos permitirá crear objetos como son los triggers, los join que se habilitarán para poder tener un mayor control sobre los datos que serán almacenados, esto nos permitirá que la base de datos sea confiable y a la vez estable y que por tanto tenga un buen desempeño en el valor de los mismo (Vea figura 4.3.10).

➤ **Creación de un procedimiento almacenado.**

Para la creación de un procedimiento almacenado se debe desplegar el árbol de opciones de nuestra base de datos, ahí aparecerá la opción de procedimientos almacenados, sobre esta opción dar clic derecho y nuevo procedimiento almacenado. Se abrirá la pantalla de procedimientos almacenados, en donde podremos definir uno nuevo. (Vea figura 4.3.11).



4.4 Diseño y construcción del front-end.

El concepto de front-end puede tener varias interpretaciones en función del contexto bajo el cual se esté tratando, pero de manera resumida y para los fines que en el presente trabajo se persiguen se puede decir que: “el front-end es la parte del software que interactúa con todos los usuarios del sistema”, la idea general es que el front-end sea el responsable de recolectar los datos de entrada del usuario, esta tarea puede ser desempeñada de muchas y variadas formas, pero finalmente lo más importante es que deben ser entregados de modo tal que el back-end pueda hacer uso de ellos.

En el presente trabajo se hace uso de Visual Basic 6 para el desarrollo y la construcción de los módulos que integran al Sistema que nos permitirá registrar y controlar las fallas técnicas de los POS en las taquillas del Sistema de Transporte Colectivo, se hace uso de esta herramienta ya que entre otras características, proporciona herramientas adecuadas para el diseño y desarrollo de la interfaz gráfica de usuario de una manera sencilla y amigable.

4.4.1 Pantalla de ingreso.

La primera ventana que se mostrará en pantalla y que nos permitirá identificar el tipo de usuario (administrador o capturista) que está por entrar al Sistema es la siguiente (figura 4.4.1.1):



Figura 4.4.1.1 Pantalla inicial.

Esta ventana nos muestra dos campos para ingresar un login o usuario y otro para ingresar una contraseña o password por medio de estos dos campos se puede tener identificado el tipo de usuario que ha de ingresar al Sistema, también nos muestra dos botones uno para entrar al Sistema ya que se han proporcionado un usuario y un password y otro para salir o cerrar esta ventana en caso necesario, en caso de ingresar un usuario o un password incorrecto de mostrará un mensaje en pantalla como el siguiente (figura 4.4.1.2).



Figura 4.4.1.2 Ventana con mensaje de error por ingreso de usuario o password incorrecto.

Una vez que se ha ingresado al Sistema existen dos posibles ventanas que pueden ser mostradas en función del tipo de usuario que se identifique, en la siguiente figura se muestra la ventana para el usuario administrador (figura 4.4.1.3):

Figura 4.4.1.3 Ventana mostrada al usuario administrador.

Por otra parte si se registra un usuario sin privilegios o capturista la ventana mostrada será la siguiente (figura 4.4.1.4).

Figura 4.4.1.4 Ventana que se muestra a un usuario capturista.

Como se puede observar para el administrador se muestran dos módulos adicionales (CONSULTA DE ATENDIDOS y REPORTE), pero los otros dos son los mismos para ambos tipos de usuario, primero se da una explicación de los módulos comunes y posteriormente se explican los módulos que solo puede ver el administrador.



4.4.2 Módulo CAPTURA DE DATOS.

El módulo para captura de datos es el siguiente (figura 4.4.2.1), a través de él los técnicos del Sistema de Transporte Colectivo ingresan los datos de las fallas al sistema:

Nº. de Solicitud <input type="text"/>	Fecha de Reporte 24/08/2009	Hora de Reporte 06:22:42 p.m.	Estatus: PENDIENTE
Expediente: <input type="text"/>	Nombre de quien reporta: <input type="text"/>		Cargo: <input type="text"/>
DATOS DE LA TAQUILLA QUE PRESENTA LA FALLA			
Nº. de Taquilla: <input type="text"/>	Turno: PRIMERO	P.O.S. <input type="text"/>	Línea: 1
Estación: <input type="text"/>			
Tipo de Falla: Módulo CSC fuera de servicio			
Solución: Reconexión del CSC			
DATOS DEL PERSONAL DE SOPORTE			
Nombre del Asesor: ERICK SIERRA AGUILAR		Nombre de Técnico: ERICK SIERRA AGUILAR	
Turno: PRIMERO	Tipo de Atención: Telefónica	Fecha de Atención: 24/08/2009	Hora de Atención: 06:22:21 p.m.
DATOS DEL PERSONAL DE TAQUILLA			
Expediente: <input type="text"/>	Nombre: <input type="text"/>	Turno: PRIMERO	

Figura 4.4.2.1 Módulo de captura de datos.

A continuación una breve descripción de cada uno de los campos que se observan este módulo:

No. de Solicitud: Este campo es llenado de manera automática al guardar el reporte actual y es un número consecutivo.



Fecha de Reporte: Al igual que el campo anterior, es asignado de manera automática con la fecha real del sistema, para llevar un control fiable en cuanto al tiempo de atención de la falla en cuestión.

Hora de Reporte: Es un campo complementario de la Fecha de reporte y que no puede ser llenado de forma manual, es decir, no puede ser manipulado por el personal del área por fines de fiabilidad de la información.

Expediente: En este campo se captura el expediente de la supervisora que reporta la falla y es un campo que no puede estar nulo.

Nombre de quien reporta: Este campo es llenado de forma automática y está ligado con el número de expediente.

Cargo: Este campo puede adoptar el valor "supervisora" o "taquillera" y está ligado al número de expediente capturado, pero en general se reciben reportes sólo de supervisora.

En la sección "DATOS DE LA TAQUILLA QUE PRESENTA LA FALLA":

No de Taquilla: En este campo como lo dice su nombre se captura el número de la taquilla que presenta la falla para poder llevar un mejor control y así saber cuales POS son los que presentan más fallas.

Turno: Este campo nos permite llevar un control sobre cual de los tres turnos presentan más fallas.

P.O.S: Este es un campo complementario del No de Taquilla el cual no es necesario llenar manualmente y nos permite saber el No de pos que corresponde a cada taquilla.



Línea: En este campo al igual que el anterior es llenado automáticamente y este nos permite identificar en cual de las 11 líneas existe el problema y a su vez identificar las líneas más conflictivas en este aspecto.

Estación: Este campo es llenado de manera automática ya que depende directamente del número de taquilla y refleja el nombre de la estación en la cual se registra la falla.

Tipo de Falla: Este campo puede ser manipulado por el personal de Soporte Técnico colocando el tipo de falla que presenta el POS en cuestión, esto nos sirve para detectar las fallas más frecuentes.

Solución: En este campo se coloca el tipo de solución que se dio a la falla para saber el trabajo que se realizó y llevar un control de las soluciones más frecuentes con fines de capacitación.

En la sección “DATOS DEL PERSONAL DE SOPORTE”:

Nombre del Asesor: En este campo se captura el nombre de la persona del área de Soporte que recibe la llamada de reporte.

Nombre del Técnico: En este campo se registra el nombre de la persona que da solución a la falla independientemente de si ésta fue telefónica o presencial.

Turno: Es el turno del técnico que da solución, esto es solo con fines estadísticos.

Tipo de atención: En este campo se registra si la solución a la falla es telefónica o se acudió a la taquilla para lograr el regreso a operación del POS, además existe la posibilidad de que la solución sea una asesoría que es cuando la llamada para dar solución se extiende por más de 10 minutos esto también con fines estadísticos.



Fecha de atención: Aquí se coloca la fecha en que el POS regresa a operación, es decir, cuando la falla queda resuelta, aunque en la gran mayoría de los casos es el mismo día a aceptación de cuando se necesitan refacciones que no existen en el momento.

Hora de atención: En este campo se registra la hora de atención esto con la finalidad de llevar un control con referencia a la capacidad de respuesta del área ante las fallas.

En la sección “DATOS DEL PERSONAL DE TAQUILLA”:

Expediente: Es el expediente de la taquillera que se encuentra en la taquilla al momento de dar solución.

Nombre: Aquí se registra el nombre de la taquillera que se encontraba en la taquilla cuando el técnico dio solución.

Turno: Es el turno de taquilla en que se dio solución.

4.4.3 Módulo CONSULTA DE PENDIENTES.

Este módulo es únicamente de visualización para verificar si existen fallas por solucionar, este módulo es funcional para el coordinador del área de Soporte Técnico con fines de la entrega de reportes o solicitud de refacciones.

Para el personal en general es funcional en especial en los cambios de turno ya que les permite dar atención canalizando al personal y el material prudente para dar seguimiento y solución a los pendientes, la figura 4.4.3.1 muestra la ventana que se mostrará en la pantalla.



CAPTURA DE DATOS **CONSULTA DE ATENDIDOS** CONSULTA DE PENDIENTES REPORTE

Del: 24/08/2009 **Al:** 24/08/2009 **Taquilla:** **Turno:** PRIMERO

Tipo de Falla: Módulo CSC fuera de servicio

Solución: Reconexión del CSC **Línea:** 1 **Expediente Taquillera:**

Solicitud	Fecha	Turno	Exp. Taq.	Taquilla	Estación	Fecha Atn.	Fallas	Soluc
-----------	-------	-------	-----------	----------	----------	------------	--------	-------

Total: Total asistidas:

Figura 4.4.4.1 Modulo de Consulta de Atendidos.

Las consulta se pueden realizar por un rango de fechas por medio de los campos “Del” y “Al” que reciben fechas inicial y final para la obtención de reportes.

Al colocar un dato en el campo “Taquilla” se muestran únicamente las fallas que se han presentado en la taquilla que se capture en este campo.

En los campos “Turno”, “Tipo de falla”, “Solución” y “Línea” se encuentra unos cuadros de selección al lado izquierdo, al habilitar uno o más de estos cuadros se realiza un filtrado por cada uno o más de estos criterios de selección.



Finalmente se puede realizar una consulta por expediente de taquillera, esto con la finalidad de llevar un control estadístico del personal de taquilla que más problemas tiene con el POS.

4.4.5 Módulo REPORTE.

En la figura 4.4.5.1 se puede observar la ventana que el sistema mostrará en pantalla.

Figura 4.4.5.1 Módulo Reporte.

El presente módulo tiene la finalidad de presentar en pantalla un reporte estadístico como el que se muestra en la figura 4.4.5.2 que representa estadísticamente el desempeño del personal del área de soporte técnico.

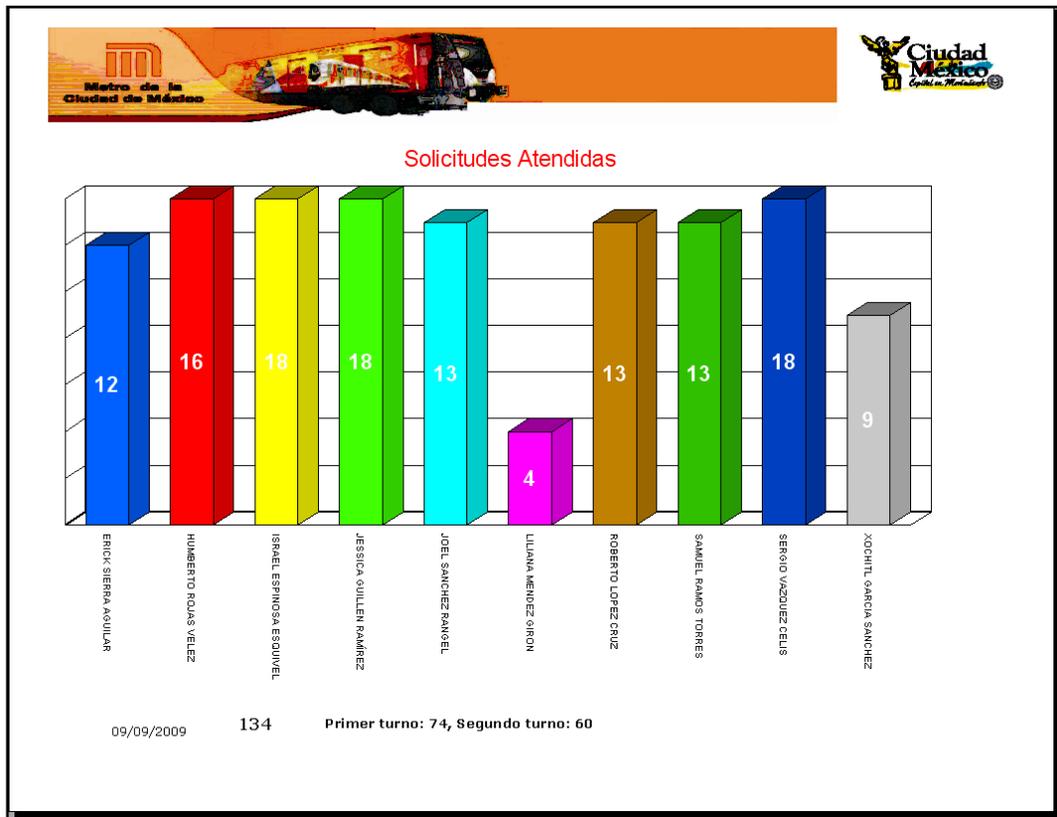


Figura 4.4.5.2 Gráfica estadística del desempeño del personal de Soporte Técnico.



4.5 Pruebas del sistema, afinación y mantenimiento de la base de datos.

Una vez que se tiene construido el Back-end y el Front-end, que integrará nuestro sistema, es indispensable realizar diversas pruebas al sistema para verificar su funcionamiento y así poder evitar cualquier error durante la puesta en operación, así mismo es necesario realizar la afinación de la base de datos, así como realizar los diferentes tipos de mantenimiento de los cuales hablaremos más adelante, para garantizar el buen funcionamiento de la misma.

4.5.1. Pruebas del Sistema.

Una de las últimas fases del ciclo de vida antes de entregar un programa para su explotación, es la fase de pruebas.

La fase de pruebas añade valor al producto que se maneja: todos los programas tienen errores y la fase de pruebas los descubre; ese es el valor que añade.

El objetivo de la fase de pruebas es destapar errores, es decir probar un programa es ejercitarlo con la peor intención a fin de encontrarle fallos.

Probar un programa es someterle a todas las posible variaciones de los datos de entrada, tanto si son válidos como si no lo son.

➤ Prueba de Unidades.

La prueba de unidades se plantea a pequeña escala, y consiste en ir probando uno a uno los diferentes módulos que constituyen una aplicación.

➤ Caja blanca.

Se llama caja blanca cuando lo que se mira con lupa es el código que está ahí escrito y se intenta que falle. Quizás sea más propia la denominación de "pruebas de caja transparente". Esta prueba se subdivide en diferentes tipos:

Cobertura de segmentos.

También denominada "cobertura de sentencias". Por segmento se entiende una secuencia de sentencias sin puntos de decisión. Como el ordenador está obligado



a ejecutarlas una tras otra, es lo mismo decir que se han ejecutado todas las sentencias o todos los segmentos.

Cobertura de ramas.

Consistente en recorrer todas las posibles salidas de los puntos de decisión.

Cobertura de Condición / Decisión.

Consiste en trocear las expresiones booleanas complejas en sus componentes e intenta cubrir todos los posibles valores de cada uno de ellos.

➤ **Caja negra.**

Se dice que una prueba es de caja negra cuando prescinde de los detalles del código y se limita a lo que se ve desde el exterior. Intenta descubrir casos y circunstancias en los que el módulo no hace lo que se espera de él.

Se le conoce también como pruebas de caja opaca, pruebas funcionales, pruebas de entrada/salida y pruebas inducidas por los datos. Las pruebas de caja negra se centran en lo que se espera de un módulo, es decir, intentan encontrar casos en que el módulo no se atiene a su especificación. Por ello se denominan pruebas funcionales, y el probador se limita a suministrarle datos como entrada y estudiar la salida, sin preocuparse de lo que pueda estar haciendo el módulo por dentro.

Las pruebas de caja negra están especialmente indicadas en aquellos módulos que van a ser interfaz con el usuario (en sentido general: teclado, pantalla, ficheros, canales de comunicaciones, etc.).

Las pruebas de caja negra se apoyan en la especificación de requisitos del módulo. A la vista de los requisitos de un módulo, se sigue una técnica algebraica conocida como "clases de equivalencia". Esta técnica trata cada parámetro como un modelo algebraico donde unos datos son equivalentes a otros.

En la figura 4.5.1.1 se muestra un ejemplo de pruebas de caja negra.



Figura 4.5.1.1 Ejemplo prueba caja negra.

4.5.1.2 Pruebas de Integración.

Las 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.

4.5.1.3 Pruebas de Aceptación.

Estas pruebas las realiza el cliente. Son básicamente pruebas funcionales, sobre el sistema completo, y buscan una cobertura de la especificación de requisitos y del manual del usuario. Estas pruebas no se realizan durante el desarrollo.



4.5.1.4. Otros tipos de pruebas.

- **Recorridos** (walkthroughs). Es una técnica más aplicada en control de calidad que en pruebas. Consiste en sentar alrededor de una mesa a los desarrolladores y a una serie de críticos, leer el programa línea a línea y pedir explicaciones de todo lo que no está claro.
- **Aleatorias** (random testing). Ciertos autores consideran que la probabilidad de descubrir un error es prácticamente la misma si se hacen una serie de pruebas aleatoriamente elegidas, que si se hacen siguiendo las instrucciones dictadas por criterios de cobertura (caja negra o blanca).
- **Solidez** (robustness testing). Se prueba la capacidad del sistema para salir de situaciones embarazosas provocadas por errores en el suministro de datos.
- **Aguante** (stress testing). En ciertos sistemas es conveniente saber hasta dónde aguantan, bien por razones internas ¿hasta cuantos datos podrá procesar?, bien externas ¿es capaz de trabajar con un disco al 90%?, ¿aguanta una carga de la CPU del 90?, etc.
- **Prestaciones** (performance testing). A veces es importante el tiempo de respuesta, u otros parámetros de gasto. Típicamente nos puede preocupar cuánto tiempo le lleva al sistema procesar tantos datos, o cuánta memoria consume, o cuánto espacio en disco utiliza, o cuántos datos transfiere por un canal de comunicaciones.
- **Regresión** (regression testing). Cuando a un sistema se le hace alguna modificación o sufre alguna evolución, simplemente se prueba que las modificaciones no provocan errores dentro del mismo.
- **Mutación** (mutation testing). Es una técnica curiosa consistente en alterar ligeramente el sistema bajo pruebas introduciendo errores para averiguar si nuestra batería de pruebas es capaz de detectarlo.
- **Depuración** (debugging). Los depuradores pueden usarse para realizar inspecciones rigurosas sobre el comportamiento dinámico de los programas. El objetivo habitual es utilizarlo como consecuencia de la detección de un error.



Pruebas aplicadas al sistema de registro y control de fallas técnicas en los equipos de recarga (POS).

Las pruebas realizadas al sistema en su etapa inicial fueron la de caja blanca para examinar la estructura interna del programa, particularmente los DFD.

De las pruebas de caja negra, en lo que a la parte funcional del sistema se refiere, realizamos las pruebas particulares de errores de interface dada la cantidad de ventanas que tenemos para la interacción con el usuario.

También se realizaron pruebas de estructura de la base de datos, en donde nos encontramos con el problema de la incorrecta definición de llaves primarias, la falta de normalización de la base de datos, así como el número de tablas y datos empleados incorrectamente, lo cual elevaba el tiempo de proceso además de la dificultad que presentaba en el momento de realizar la programación. Estas pruebas de caja negra se realizaron de manera exhaustiva, hasta el momento en que no encontramos más errores.

Prácticamente todas las pruebas mencionadas en este capítulo pueden tener aplicación sobre nuestro sistema, siendo todas éstas de igual importancia y útiles en las etapas de prueba e implementación, sin embargo; por el resultado que nos arrojan y tomando en cuenta además, el gasto en tiempo, esfuerzo y dinero implicado, se decidió implementar solo las que consideramos las más adecuadas y las cuales ya fueron mencionadas.

Referente a las pruebas de validación nos aseguramos de que el software creado satisface los requerimientos del usuario ya que a cada paso del desarrollo del sistema nos preguntamos si estábamos construyendo el producto correcto. Confirmando en todo momento la respuesta positiva. Se aplicó también la verificación, asegurándonos que la nueva aplicación se está construyendo correctamente, se implementaron también las pruebas de integración, empleando



particularmente la incremental, pues al combinar módulos, debíamos probar el conjunto de éstos.

Finalmente se realizó la prueba de estrés o de volumen, necesaria al estar nuestro sistema trabajando en red; para lo cual obtuvimos resultados exitosos.

4.5.2 Afinación de la Base de datos.

El lenguaje de consulta a base de datos SQL nos permite explotar la flexibilidad y potencia de los sistemas relacionales, permitiéndonos gran variedad de operaciones sobre los mismos.

El orden de ejecución interno de una sentencia puede afectar gravemente a la eficiencia del RDBMS, por lo que se hace necesario que éste lleve a cabo una optimización antes de la ejecución de la misma.

Usualmente, el uso de índices acelera una instrucción de consulta, pero hace lenta la actualización de los datos, dependiendo del uso de la aplicación, se dará el acceso indexado o una rápida actualización de la información.

La optimización o afinación, difiere sensiblemente en cada motor de base de datos y depende de muchos factores.

Cualquiera que sea la técnica de ajuste empleada en la optimización de la base de datos, se tiene cuidado en:

- Reducir la actividad de los discos, indexando las vistas que lo justifiquen.
- Reducir el tráfico en la red.
- Reducir la utilización del CPU.



4.5.2.1 Afinación en la recuperación de datos.

La optimización del diseño de la base de datos se debe afinar desde el comienzo, dejando margen para futuras modificaciones, teniendo entre las prácticas más comunes:

- Normalización de las tablas, no sobre-normalizar y desnormalizar cuando tenga sentido.
- Analizar el plan de ejecución de los procesos.

Sobre los **queries** (consultas) utilizados, encontramos que existen argumentos de búsqueda que las acotan. El consejo para este caso es:

- Usar OLE DB para acceder datos, y evitar ODBC si se puede.
- Evitar el uso de cursores.
- Emplear los triggers con cautela, tanto INSTEAD OF como AFTER.
- Recuperar lo necesario y nunca datos de más que no se utilizarán y solo harán lenta la recuperación de información desde la base de datos.
- Utilizar StoreProcedures en vez de queries.
- Agregar, quitar o modificar los índices cuando sea posible.

4.5.2.2. Afinación respecto al uso del procesador.

Se recomienda utilizar siempre un servidor multiprocesador, con la finalidad de optimizar el uso del procesador durante el trabajo con la base de datos. De preferencia, el servidor donde se encuentre SQL server, debe ser dedicado exclusivamente para el almacenamiento de la base de datos. De esta manera se evita ejecutar otras aplicaciones o servicios en él, lo que le resta rendimiento. De hecho, el servidor debería ser un **member server** (servidor miembro) del dominio. Así mismo, Siendo la **RAM** (Random Access Memory / Memoria de Acceso Aleatorio) el primer factor determinante del desempeño, es importante utilizar toda la que se pueda comprar y que a su vez el sistema operativo requiera y soporte.



- **W2K Server:** hasta 4 procesadores y hasta 2 GB de memoria RAM.
- **W2K Advanced Server:** hasta 8 procesadores y hasta 8 GB de memoria RAM.
- **W2K Datacenter Server:** hasta 32 procesadores y hasta 64 GB de memoria RAM.

Almacenamiento

Es recomendable instalar un RAID por hardware y desactivar todo el caché de disco en las tarjetas controladoras de éstos. Se sugiere:

- RAID 5 archivos de datos.
- RAID 0+1 archivos de log.

4.5.2.3 Utilerías auxiliares.

Existen una serie de herramientas adicionales que auxilian a mejorar el desempeño de la base de datos y afinar su utilización y se dividen en:

Herramientas principales:

- Windows Performance Monitor.
- SQL Server Profiler.

Herramientas secundarias:

- Registros de error del SQL Server.
- Registros de error del SQL Server Agent.
- Registro de sucesos del sistema operativo.
- Actividad actual del servidor (Ent. Manager).
- Comandos DBCC (actividad e integridad de la BBDD).
- sp_who / sp_lock / sp_monitor.



4.5.3 Mantenimiento de la Base de Datos.

Dentro el mantenimiento que se puede dar a una base de datos existen los siguientes:

- Preventivo: Es el óptimo, ya que sucede previo a la implementación de la base de datos, o una vez que esta ya ha sido puesta en uso, y cuida aspectos como evitar saturar el uso del CPU, memoria, uso de arreglos de discos, entre los más importantes.
- Correctivo: Tiene lugar una vez que la base ya fue implementada y se suscita algún error o complicación en su uso.
- Tecnológico: Realizado para tener al día la base de datos, en cuanto a versiones se refiere, y aplicar los mismos criterios sobre el entorno de trabajo (procesador, memoria, sistema operativo, herramientas adicionales).
- Perfectivo: Este un mantenimiento que suele dejarse de lado y tiene lugar una vez que se sucede un error o problema con la base de datos, pues es hasta ese momento que el administrador se preocupa por mejorar lo ya establecido en vez de realizarlo como parte de la prevención o mantenimiento a la base de datos.

Existen diversas herramientas que auxilian a Microsoft SQL Server 2005 en el mantenimiento a la base de datos. En la tabla 4.5.3.1 se muestra el nombre, sintaxis y función de algunas de ellas.



NOMBRE	SINTAXIS	FUNCION
sp_spaceused	sp_spaceused [[@objname =] 'objname'] [, [@updateusage =] 'updateusage']	Muestra el espacio físico que utiliza la base de datos.
sp_changedbowner	sp_changedbowner [@loginame =] 'login' [, [@map =] remap_alias_flag]	Cambia al propietario de la base de datos en la base de datos actual.
sp_renamedb	sp_renamedb [@dbname =] 'old_name' , [@newname =] 'new_name'	Cambia el nombre de la base de datos especificada.
sp_attach_db	sp_attach_db [@dbname =] 'dbname', [@filename1 =] 'filename_n' [,... 16]	Adjunta una base de datos a un servidor SQL.
sp_detach_db	sp_detach_db [@dbname=] 'database_name' [, [@skipchecks=] 'skipchecks']	Remueve las referencias en la base de datos maestra acerca de la base de datos específica.

Tabla 4.5.3.1 Herramientas para el mantenimiento de Bases de Datos en SQL.

Cambio de la ubicación física de la base de datos.

Mediante el Asistente únicamente podremos mover o copiar una base de datos de un servidor o instancia a otro, no es posible copiar o mover la base de datos maestra.

Los puntos que se deben considerar antes de mover a otro servidor o instancia una base de datos son:



- Los códigos de scripts o aplicaciones deben ser codificados nuevamente.
- La localización física del archivo en el nuevo servidor o instancia.
- Problemas de desempeño con relación a la infraestructura de red establecida en el entorno del nuevo servidor o instancia.
- La aceptación de firma inicial para uso de la base de datos.

Incremento del tamaño de la base de datos.

Cada una de las siguientes tareas pueden ser completadas por cualquiera de las siguientes opciones listadas, en todos los casos la sentencia es ALTER DATABASE.

- Incremento del tamaño físico de la base de datos.
- Agregar archivos o grupos de archivos a la base de datos.
- Agregar datos o archivos de registro de incidentes (logs).
- Cambiar el grupo de archivos por defecto.

Decremento del tamaño físico de la base de datos.

El decremento del tamaño físico de la base de datos puede lograrse al completar cualquiera de las tareas listadas a continuación:

- Reducir la base de datos (DBCC SHRINKDATABASE).
- Cambiar las opciones de reducción de la base de datos (sp_dboption).
- Reducción de un archivo, de manera individual de la base de datos (DBCC SHRINKFILE).
- Borrar datos o archivos de registro (ALTER DATABASE).

4.6 Obtención de Reportes.

Como ya se mencionó anteriormente, el sistema nos da la posibilidad de imprimir diversos reportes, en este capítulo detallaremos cuales y como obtener dichos reportes.



Modulo Captura de datos.

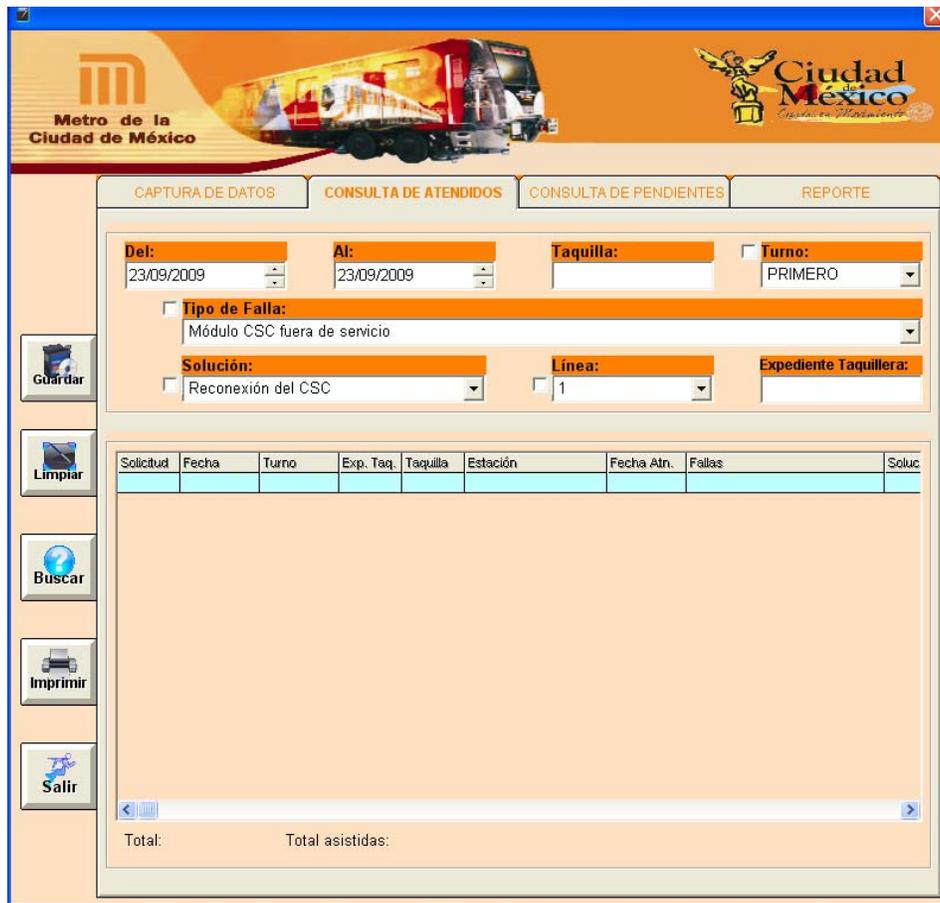
Dentro de este modulo, nos da la posibilidad de imprimir una solicitud nueva, para con ella poder atender en sitio la falla reportada.

Una vez capturada la información necesaria, debemos dar clic en el botón situado del lado derecho llamado **Imprimir**. (Vea figura 4.6.1).

Figura 4.6.1 Impresión de una solicitud de atención de falla.

Modulo Consulta de Atendidos.

En este modulo se cuenta con la posibilidad de poder consultar reportes con diferentes parámetros y/o restricciones, dependiendo de los datos que se requieran demos activar o no las casillas que tenemos por opción (Vea figura 4.6.2).



Metro de la Ciudad de México

Ciudad de México

CAPTURA DE DATOS CONSULTA DE ATENDIDOS CONSULTA DE PENDIENTES REPORTE

Del: 23/09/2009 Al: 23/09/2009 Taquilla: Turno: PRIMERO

Tipo de Falla:
Módulo CSC fuera de servicio

Solución: Línea: Expediente Taquillera:

Reconexión del CSC 1

Solicitud	Fecha	Turno	Exp. Taq.	Taquilla	Estación	Fecha Atn.	Fallas	Soluc

Total: Total asistidas:

Figura 4.6.2 Selección de los campos que se desea presente el reporte de Consulta de Atendidos.

Como se puede observar dentro de este modulo podemos consultar directamente en pantalla los datos que deseemos, y posteriormente con el botón situado del lado izquierdo llamado **Imprimir**, podemos obtener la impresión de dicho reporte. Dentro de los reportes que podemos obtener se encuentran los siguientes:

- Fallas atendidas indicando únicamente un período de tiempo (Vea figuras 4.6.3 y 4.6.4).
- Fallas atendidas indicando un período de tiempo y una taquilla en específico (Vea figura 4.6.5 y 4.6.6).
- Fallas atendidas indicando un período de tiempo y el expediente de una taquillera en específico (Vea figura 4.6.7 y 4.6.8).

- Fallas atendidas indicando un período de tiempo y el turno en la que se presentó la falla (Vea figura 4.6.9 y 4.6.10).
- Fallas atendidas indicando un periodo de tiempo y el tipo de falla (Vea figura 4.6.11 y 4.6.12).
- Fallas atendidas indicando un periodo de tiempo y la solución (Vea figura 4.6.13 y 4.6.14).
- Fallas atendidas indicando un periodo de tiempo y la línea en la que se presentó la falla (Vea figura 4.6.15 y 4.6.16).
- Fallas atendidas indicando un periodo de tiempo, turno, tipo de falla, solución y línea. (Vea figura 4.6.17 y 4.6.18).

Solicitud	Fecha	Turno	Exp. Taq.	Taquilla	Estación	Fecha Atn.	Fallas	Sc
2467	31/Ago/2009	SEGUNDO	20956	1062	MOCTEZUMA	22/Sep/2009	Falla de la impresora	Ce
2471	01/Sep/2009	PRIMERO	22723	7022	MIXCOAC	01/Sep/2009	No enciende el equipo	Nc
2472	01/Sep/2009	PRIMERO	10338	2201	POPOTLA	01/Sep/2009	Pantalla incorrecta	Ci
2473	01/Sep/2009	PRIMERO	31093	1062	MOCTEZUMA	01/Sep/2009	Pantalla incorrecta	Re
2474	01/Sep/2009	PRIMERO	17639	2141	BELLAS ARTES L-2	01/Sep/2009	Falla de la impresora	As
2475	01/Sep/2009	SEGUNDO	23072	1111	ISABEL LA CATOLICA	01/Sep/2009	No enciende el equipo	Nc
2476	01/Sep/2009	SEGUNDO	27739	3102	CENTRO MEDICO L-3	01/Sep/2009	Pantalla incorrecta	Er
2477	01/Sep/2009	SEGUNDO	30241	2121	ZOCALO	01/Sep/2009	Pantalla incorrecta	Er
2478	02/Sep/2009	PRIMERO	28259	2231	PANTEONES	02/Sep/2009	No enciende el equipo	Mk
2479	02/Sep/2009	PRIMERO	10338	2202	POPOTLA	02/Sep/2009	No enciende el equipo	Nc
2480	02/Sep/2009	PRIMERO	18586	3041	YIVEROS	02/Sep/2009	No enciende el equipo	Nc
2481	02/Sep/2009	PRIMERO	15487	B051	MUZQUIZ	02/Sep/2009	Bloqueo de cuenta	De
2482	02/Sep/2009	PRIMERO	25119	1072	SAN LAZARO	03/Sep/2009	No enciende el equipo	Mk
2483	02/Sep/2009	PRIMERO	15917	5061	EDUARDO MOLINA	02/Sep/2009	Pantalla incorrecta	Ci
2484	02/Sep/2009	PRIMERO	25545	2243	CUATRO CAMINOS	02/Sep/2009	Falla de la impresora	Pa
2485	02/Sep/2009	PRIMERO	5650	B132	ROMERO RUBIO	02/Sep/2009	Falla de la impresora	As

Figura 4.6.3 Visualización en pantalla del reporte de fallas atendidas sin filtros.



Listado de solicitudes atendidas por fecha

1 de 7 Total: 186 100% 186 de 186

Metro de la Ciudad de México

Ciudad México

GERENCIA DE ORGANIZACIÓN Y SISTEMAS

SOPORTE TÉCNICO A P.O.S.

Miércoles, 23 de Septiembre de 2009

Nº de Solicitud	Fecha de Falla	Turno	Nº de Taquilla	Estación	Fecha de Atención	Falla	Solución
2407	31-Ago-09	SEGUNDO	1063	MÓCTEZUMA	23-Sep-09	Falla de la impresora	Cambio de impresora
2471	1-Sep-09	PRIMERO	7022	MIGÜEL	1-Sep-09	No enciende el equipo	No break apagado
2402	1-Sep-09	PRIMERO	2201	POPOTLA	1-Sep-09	Pantalla incorrecta	Reinicio de P.O.S
2470	1-Sep-09	PRIMERO	1063	MÓCTEZUMA	1-Sep-09	Pantalla incorrecta	Reinicio de P.O.S
2474	1-Sep-09	PRIMERO	2141	BELLAS ARTES L-2	1-Sep-09	Falla de la impresora	Apuesta
2435	1-Sep-09	SEGUNDO	1111	ISABEL LA CATOLICA	1-Sep-09	No enciende el equipo	No break apagado
2476	1-Sep-09	SEGUNDO	3102	CENTRO MEDICO L-3	1-Sep-09	Pantalla incorrecta	Error de Conexión en la Base de Datos
2477	1-Sep-09	SEGUNDO	2121	ZÓCALO	1-Sep-09	Pantalla incorrecta	Error de Conexión en la Base de Datos
2478	2-Sep-09	PRIMERO	2231	PANTEONES	2-Sep-09	No enciende el equipo	Mantenimiento a la Fuente de Poder
2479	2-Sep-09	PRIMERO	2202	POPOTLA	2-Sep-09	No enciende el equipo	No break apagado
2480	2-Sep-09	PRIMERO	2041	VIVEROS	2-Sep-09	No enciende el equipo	No break apagado
2481	2-Sep-09	PRIMERO	8051	MIZQUIC	2-Sep-09	Bloqueo de cuenta	Desbloqueo de cuenta
2482	2-Sep-09	PRIMERO	1072	SAN LACARO	2-Sep-09	No enciende el equipo	Mantenimiento a la fuente de Poder
2483	2-Sep-09	PRIMERO	9201	EDUARDO MOLINA	2-Sep-09	Pantalla incorrecta	Cierre de pantallas
2484	2-Sep-09	PRIMERO	2240	CUATRO CAMINOS	2-Sep-09	Falla de la impresora	Apuesta mal colocada
2485	2-Sep-09	PRIMERO	8122	ROMERO RUBIO	2-Sep-09	Falla de la impresora	Apuesta
2486	2-Sep-09	SEGUNDO	1112	ISABEL LA CATOLICA	2-Sep-09	Pantalla incorrecta	Reinicio de P.O.S
2487	2-Sep-09	SEGUNDO	1004	ZAPATA	2-Sep-09	Equipó trabado	Reinicio de P.O.S

Figura 4.6.4 Impresión del reporte de fallas atendidas sin filtros.

Metro de la Ciudad de México

Ciudad México

CAPTURA DE DATOS CONSULTA DE ATENDIDOS CONSULTA DE PENDIENTES REPORTE

Del: 01/09/2009 Al: 23/09/2009 Taquilla: 1011 Turno: PRIMERO

Tipo de Falla: Módulo CSC fuera de servicio

Solución: Reconexión del CSC Linea: 1 Expediente Taquillera:

Solicitud	Fecha	Turno	Exp. Taq.	Taquilla	Estación	Fecha Atn.	Fallas	Soluc.
2656	23/Sep/2009	SEGUNDO	9605	1011	PANTITLAN L-1	23/Sep/2009	Equipó trabado	Reinc.

Total: 1 Total asistidas: 0

Figura 4.6.5 Visualización en pantalla de las fallas atendidas indicando la taquilla.



Listado de solicitudes atendidas por fecha

1 de 1 100% Total:1 100% 1 de 1




GERENCIA DE ORGANIZACIÓN Y SISTEMAS

SOPORTE TÉCNICO A P.O.S.
Miércoles, 23 de Septiembre de 2009

Nº de Solicitud	Fecha de Solicitud	Turno	Nº de Taquilla	Estación	Fecha de Atención	Falla	Solución
2.856	23-Sep-09	SEGUNDO	1011	PANTITLAN L-1	23-Sep-09	Equipó trabado	Reinicio de P.O.S

Figura 4.6.6 Impresión del reporte de fallas atendidas indicando la taquilla.




CAPTURA DE DATOS
CONSULTA DE ATENDIDOS
CONSULTA DE PENDIENTES
REPORTE

Del: 01/09/2009 Al: 23/09/2009 Taquilla: Turno: TERCERO

Tipo de Falla: Módulo CSC fuera de servicio

Solución: Reparación del CSC Línea: 1 Expediente Taquillera: 27110

Solicitud	Fecha	Turno	Exp. Taq.	Taquilla	Estación	Fecha Atn.	Fallas	Soluc
2636	21/Sep/2009	SEGUNDO	27110	3032	MIGUEL ANGEL DE Q.	21/Sep/2009	Equipó trabado	Reinic

Guardar
Limpiar
Buscar
Imprimir

Figura 4.6.7 Visualización en pantalla de las fallas atendidas indicando el expediente de la taquillera.



Listado de solicitudes atendidas por fecha

1 de 1 100% Total: 1 100% 1 de 1

GERENCIA DE ORGANIZACIÓN Y SISTEMAS

SOPORTE TÉCNICO A P.O.S.

Miércoles, 23 de Septiembre de 2009

Nº de Solicitud	Fecha de Solicitud	Turno	Nº de Taquilla	Estación	Fecha de Atención	Falla	Solución
2,636	21-Sep-09	SEGUNDO	3032	MIGUEL ANGEL DE O.	21-Sep-09	Equipo trabado	Reinicio de P.O.S

Figura 4.6.8 Impresión del reporte de fallas atendidas indicando el expediente de la taquillera.

CAPTURA DE DATOS
CONSULTA DE ATENDIDOS
CONSULTA DE PENDIENTES
REPORTE

Del: 01/09/2009 Al: 23/09/2009 Taquilla: Turno: TERCERO

Tipo de Falla: Módulo CSC fuera de servicio

Solución: Reconexión del CSC Linea: 1 Expediente Taquillera:

Solicitud	Fecha	Turno	Exp. Taq.	Taquilla	Estación	Fecha Atr.	Fallas	Sc
2528	08/Sep/2009	TERCERO	20238	1181	JUANACATLAN	08/Sep/2009	Falla de teclado	Bli
2529	08/Sep/2009	TERCERO	32062	3101	CENTRO MEDICO L-3	08/Sep/2009	No enciende el equipo	Fs
2543	09/Sep/2009	TERCERO	27969	1102	PINO SUAREZ L-1	09/Sep/2009	Equipo trabado	Re
2556	10/Sep/2009	TERCERO	17820	1162	SEVILLA	10/Sep/2009	Falla de teclado	Re
2557	10/Sep/2009	TERCERO	15981	9061	JAMAICA L-9	10/Sep/2009	Módulo CSC fuera de servicio	Re
2584	14/Sep/2009	TERCERO	31705	7081	POLANCO	14/Sep/2009	No enciende el equipo	Fe
2589	14/Sep/2009	TERCERO	30590	1111	ISABEL LA CATOLICA	14/Sep/2009	Falla de teclado	Re
2591	14/Sep/2009	TERCERO	16521	6091	AZCAPOTZALCO	14/Sep/2009	Bloqueo de cuenta	De
2618	17/Sep/2009	TERCERO	26359	2022	GENERAL ANAYA	18/Sep/2009	No enciende el equipo	Mk
2623	18/Sep/2009	TERCERO	26359	2022	GENERAL ANAYA	18/Sep/2009	No enciende el equipo	Re
2625	18/Sep/2009	TERCERO	9897	9071	CHABACANO L-9	18/Sep/2009	Falla de recarga	As
2626	18/Sep/2009	TERCERO	18590	2081	VIADUCTO L-2	18/Sep/2009	Falla de recarga	As
2637	21/Sep/2009	TERCERO	30242	3142	JUAREZ	21/Sep/2009	Bloqueo de cuenta	De
2638	21/Sep/2009	TERCERO	13887	1112	ISABEL LA CATOLICA	21/Sep/2009	Falla del display	Re
2639	21/Sep/2009	TERCERO	30590	1111	ISABEL LA CATOLICA	21/Sep/2009	Equipo trabado	Re
2649	22/Sep/2009	TERCERO	14779	2061	VILLA DE CORTES	22/Sep/2009	Bloqueo de cuenta	De

Figura 4.6.9 Visualización en pantalla de las fallas atendidas indicando el turno.



Listado de solicitudes atendidas por fecha

1 de 1 Total:16 100% 16 de 16




GERENCIA DE ORGANIZACIÓN Y SISTEMAS

SOPORTE TÉCNICO A P.O.S.

Miércoles, 23 de Septiembre de 2009

Nº de Solicitud	Fecha de Solicitud	Turno	Nº de Taquilla	Estación	Fecha de Atención	Falla	Solución
2,528	8-Sep-09	TERCERO	1181	JUANACATLAN	8-Sep-09	Falla de teclado	Bloq Num desactivado
2,529	8-Sep-09	TERCERO	3101	CENTRO MEDICO L-3	8-Sep-09	No enciende el equipo	Falla de luz (BT)
2,543	9-Sep-09	TERCERO	1102	PINO SUAREZ L-1	9-Sep-09	Equipó trabado	Reinicio de P O S
2,556	10-Sep-09	TERCERO	1162	SEVILLA	10-Sep-09	Falla de teclado	Reinicio de P O S
2,557	10-Sep-09	TERCERO	9061	JAMAICA L-9	10-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2,584	14-Sep-09	TERCERO	7081	POLANCO	14-Sep-09	No enciende el equipo	Falla de luz (BT)
2,589	14-Sep-09	TERCERO	1111	ISABEL LA CATOLICA	14-Sep-09	Falla de teclado	Reinicio de P O S
2,591	14-Sep-09	TERCERO	6091	AZCAPOTZALCO	14-Sep-09	Bloqueo de cuenta	Desbloqueo de cuenta
2,618	17-Sep-09	TERCERO	2022	GENERAL ANAYA	18-Sep-09	No enciende el equipo	Mantenimiento a la Fuente de Poder
2,623	18-Sep-09	TERCERO	2022	GENERAL ANAYA	18-Sep-09	No enciende el equipo	Reparacion Fuente de Poder
2,625	18-Sep-09	TERCERO	9071	CHABACANO L-9	18-Sep-09	Falla de recarga	Asesoria
2,626	18-Sep-09	TERCERO	2081	VIADUCTO L-2	18-Sep-09	Falla de recarga	Asesoria

Figura 4.6.10 Impresión del reporte de fallas atendidas indicando el turno.




CAPTURA DE DATOS CONSULTA DE ATENDIDOS CONSULTA DE PENDIENTES REPORTE

Del: 01/09/2009 Al: 23/09/2009 Taquilla: Turno: TERCERO

Tipo de Falla: Módulo CSC fuera de servicio

Solución: Reconexión del CSC Línea: 1 Expediente Taquilla:

Solicitud	Fecha	Turno	Exp. Taq.	Taquilla	Estación	Fecha Atn.	Fallas	Soluc
2494	03/Sep/2009	PRIMERO	26056	8151	DOCTORES	03/Sep/2009	Módulo CSC fuera de servicio	Recor
2496	03/Sep/2009	PRIMERO	20822	1093	MERCED	03/Sep/2009	Módulo CSC fuera de servicio	Recor
2499	04/Sep/2009	PRIMERO	27729	8051	ATLALILCO	04/Sep/2009	Módulo CSC fuera de servicio	Recor
2513	07/Sep/2009	PRIMERO	9865	7121	CAMARONES	07/Sep/2009	Módulo CSC fuera de servicio	Recor
2514	07/Sep/2009	PRIMERO	27156	1121	SALTO DEL AGUA L-1	07/Sep/2009	Módulo CSC fuera de servicio	Recor
2518	07/Sep/2009	SEGUNDO	3172	8092	IZTACALCO	07/Sep/2009	Módulo CSC fuera de servicio	Recor
2545	10/Sep/2009	PRIMERO	23047	A091	LOS REYES	10/Sep/2009	Módulo CSC fuera de servicio	Recor
2547	10/Sep/2009	PRIMERO	26064	2022	GENERAL ANAYA	10/Sep/2009	Módulo CSC fuera de servicio	Recor
2557	10/Sep/2009	TERCERO	15981	9061	JAMAICA L-9	10/Sep/2009	Módulo CSC fuera de servicio	Recor
2566	11/Sep/2009	SEGUNDO	31316	1042	BOULEVARD PTO. A.	11/Sep/2009	Módulo CSC fuera de servicio	Recor
2597	15/Sep/2009	PRIMERO	5639	1042	BOULEVARD PTO. A.	15/Sep/2009	Módulo CSC fuera de servicio	Camb
2599	15/Sep/2009	PRIMERO	17877	B142	R. FLORES MAGON	15/Sep/2009	Módulo CSC fuera de servicio	Repar
2621	18/Sep/2009	SEGUNDO	14316	3161	GUERRERO L-3	18/Sep/2009	Módulo CSC fuera de servicio	Recor
2630	21/Sep/2009	PRIMERO	8477	A081	SANTA MARTA	21/Sep/2009	Módulo CSC fuera de servicio	Recor
2648	22/Sep/2009	SEGUNDO	28779	9091	CENTRO MEDICO L-9	23/Sep/2009	Módulo CSC fuera de servicio	Repar

Figura 4.6.11 Visualización en pantalla de las fallas atendidas indicando el tipo de falla.



Listado de solicitudes atendidas por fecha

1 de 1 100% Total: 15 100% 15 de 15

SOPORTE TÉCNICO A P.O.S.
Miércoles, 23 de Septiembre de 2009

Nº de Solicitud	Fecha de Solicitud	Turno	Nº de Taquilla	Estación	Fecha de Atención	Falla	Solución
2,494	3-Sep-09	PRIMERO	8151	DOCTORES	3-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2,496	3-Sep-09	PRIMERO	1093	MERCED	3-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2,499	4-Sep-09	PRIMERO	8051	ATLALILCO	4-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2,513	7-Sep-09	PRIMERO	7121	CAMARONES	7-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2,514	7-Sep-09	PRIMERO	1121	SALTO DEL AGUA L-1	7-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2,518	7-Sep-09	SEGUNDO	8092	IZTACALCO	7-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2,546	10-Sep-09	PRIMERO	A091	LOS REYES	10-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2,547	10-Sep-09	PRIMERO	2022	GENERAL ANAYA	10-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2,557	10-Sep-09	TERCERO	9061	JAMAICA L-9	10-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2,566	11-Sep-09	SEGUNDO	1042	BOULEVARD PTO. A.	11-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2,597	15-Sep-09	PRIMERO	1042	BOULEVARD PTO. A.	15-Sep-09	Módulo CSC fuera de servicio	Cambio del cable del CSC
2,599	15-Sep-09	PRIMERO	8142	R. FLORES MAGON	15-Sep-09	Módulo CSC fuera de servicio	Reparación del CSC

Figura 4.6.12 Impresión del reporte de fallas atendidas indicando el tipo de falla.

Metro de la Ciudad de México

CAPTURA DE DATOS CONSULTA DE ATENDIDOS CONSULTA DE PENDIENTES REPORTE

Del: 01/09/2009 Al: 23/09/2009 Taquilla: Turno: TERCERO

Tipo de Falla: Módulo CSC fuera de servicio

Solución: Reparación del CSC Línea: 1 Expediente Taquillera:

Solicitud	Fecha	Turno	Exp. Taq.	Taquilla	Estación	Fecha Atn.	Fallas	Soluc
2599	15/Sep/2009	PRIMERO	17877	B142	R. FLORES MAGON	15/Sep/2009	Módulo CSC fuera de servicio	Repar
2648	22/Sep/2009	SEGUNDO	28779	9091	CENTRO MEDICO L-9	23/Sep/2009	Módulo CSC fuera de servicio	Repar

Guardar Limpiar Buscar Imprimir Salir

Figura 4.6.13 Visualización en pantalla de las fallas atendidas indicando la solución.



Listado de solicitudes atendidas por fecha

1 de 1 100% Total:2 100% 2 de 2

GERENCIA DE ORGANIZACIÓN Y SISTEMAS

SOPORTE TÉCNICO A P.O.S.

Miércoles, 23 de Septiembre de 2009

Nº de Solicitud	Fecha de Solicitud	Turno	Nº de Taquilla	Estación	Fecha de Atención	Falla	Solución
2,599	15-Sep-09	PRIMERO	B142	R. FLORES MAGON	15-Sep-09	Módulo CSC fuera de servicio	Reparación del CSC
2,648	22-Sep-09	SEGUNDO	9091	CENTRO MEDICO L-9	23-Sep-09	Módulo CSC fuera de servicio	Reparación del CSC

Figura 4.6.14 Impresión del reporte de fallas atendidas indicando la solución.

CAPTURA DE DATOS
CONSULTA DE ATENDIDOS
CONSULTA DE PENDIENTES
REPORTE

Del: 01/09/2009 Al: 23/09/2009 Taquilla: Turno: TERCERO

Tipo de Falla: Módulo CSC fuera de servicio

Solución: Reparación del CSC Línea: 1 Expediente Taquillera:

Solicitud	Fecha	Turno	Exp. Taq.	Taquilla	Estación	Fecha Atn.	Fallas	Sc
2467	31/Ago/2009	SEGUNDO	20956	1062	MOCTEZUMA	22/Sep/2009	Falla de la impresora	Ce
2473	01/Sep/2009	PRIMERO	31093	1062	MOCTEZUMA	01/Sep/2009	Pantalla incorrecta	Re
2475	01/Sep/2009	SEGUNDO	23072	1111	ISABEL LA CATOLICA	01/Sep/2009	No enciende el equipo	Nc
2482	02/Sep/2009	PRIMERO	25119	1072	SAN LAZARO	03/Sep/2009	No enciende el equipo	Me
2486	02/Sep/2009	SEGUNDO	23072	1112	ISABEL LA CATOLICA	02/Sep/2009	Pantalla incorrecta	Re
2489	02/Sep/2009	PRIMERO	25119	1072	SAN LAZARO	02/Sep/2009	No enciende el equipo	Re
2496	03/Sep/2009	PRIMERO	20822	1093	MERCED	03/Sep/2009	Módulo CSC fuera de servicio	Re
2497	03/Sep/2009	SEGUNDO	11772	1191	TACUBAYA L-1	03/Sep/2009	No enciende el equipo	Re
2508	07/Sep/2009	PRIMERO	25402	1093	MERCED	08/Sep/2009	No enciende el equipo	Me
2514	07/Sep/2009	PRIMERO	27156	1121	SALTO DEL AGUA L-1	07/Sep/2009	Módulo CSC fuera de servicio	Re
2515	07/Sep/2009	PRIMERO	9534	1021	ZARAGOZA	07/Sep/2009	Falla de mouse	Re
2520	07/Sep/2009	PRIMERO	25402	1093	MERCED	07/Sep/2009	No enciende el equipo	Re
2528	08/Sep/2009	TERCERO	20238	1181	JUANACATLAN	08/Sep/2009	Falla de teclado	Bl
2543	09/Sep/2009	TERCERO	27969	1102	PINO SUAREZ L-1	09/Sep/2009	Equipó trabado	Re
2552	10/Sep/2009	PRIMERO	24391	1192	TACUBAYA L-1	10/Sep/2009	No enciende el equipo	Re
2556	10/Sep/2009	TERCERO	17820	1162	SEVILLA	10/Sep/2009	Falla de teclado	Re

Figura 4.6.15 Visualización en pantalla de las fallas atendidas indicando la línea.



Listado de solicitudes atendidas por fecha

1 de 2 100% Total: 32 100% 32 de 32

Metro de la Ciudad de México

Ciudad México Capital en Movimiento

GERENCIA DE ORGANIZACIÓN Y SISTEMAS

SOPORTE TÉCNICO A P.O.S.
Miércoles, 23 de Septiembre de 2009

Nº de Solicitud	Fecha de Solicitud	Turno	Nº de Taquilla	Estación	Fecha de Atención	Falla	Solución
2.467	31-Ago-09	SEGUNDO	1062	MOCTEZUMA	22-Sep-09	Falla de la impresora	Cambio de Impresora
2.473	1-Sep-09	PRIMERO	1062	MOCTEZUMA	1-Sep-09	Pantalla incorrecta	Reinicio de P.O.S
2.475	1-Sep-09	SEGUNDO	1111	ISABEL LA CATOLICA	1-Sep-09	No enciende el equipo	No break apagado
2.482	2-Sep-09	PRIMERO	1072	SAN LAZARO	3-Sep-09	No enciende el equipo	Mantenimiento a la Fuente de Poder
2.486	2-Sep-09	SEGUNDO	1112	ISABEL LA CATOLICA	2-Sep-09	Pantalla incorrecta	Reinicio de P.O.S
2.489	2-Sep-09	PRIMERO	1072	SAN LAZARO	2-Sep-09	No enciende el equipo	Reparación Fuente de Poder
2.496	3-Sep-09	PRIMERO	1093	MERCED	3-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2.497	3-Sep-09	SEGUNDO	1191	TACUBAYAL-1	3-Sep-09	No enciende el equipo	Reparación No Break
2.508	7-Sep-09	PRIMERO	1093	MERCED	8-Sep-09	No enciende el equipo	Mantenimiento a la Fuente de Poder
2.514	7-Sep-09	PRIMERO	1121	SALTO DEL AGUA L-1	7-Sep-09	Módulo CSC fuera de servicio	Reconexión del CSC
2.515	7-Sep-09	PRIMERO	1021	ZARAGOZA	7-Sep-09	Falla de mouse	Reconexión de mouse
2.520	7-Sep-09	PRIMERO	1093	MERCED	7-Sep-09	No enciende el equipo	Reparación Fuente de Poder
2.528	8-Sep-09	TERCERO	1181	JUANACATLAN	8-Sep-09	Falla de teclado	Bloq Num desactivado

Figura 4.6.14 Impresión del reporte de fallas atendidas indicando la línea.

Metro de la Ciudad de México

Ciudad México Capital en Movimiento

CAPTURA DE DATOS CONSULTA DE ATENDIDOS CONSULTA DE PENDIENTES REPORTE

Del: 01/01/2009 Al: 23/09/2009 Taquilla: Turno: TERCERO

Tipo de Falla: No enciende el equipo

Solución: Reparacion Fuente de Poder Línea: 2 Expediente Taquillera:

Guardar Limpiar Buscar Imprimir Salir

Solicitud	Fecha	Turno	Exp. Taq.	Taquilla	Estación	Fecha Atn.	Fallas	Soluc
2446	27/Ago/2009	TERCERO	10486	2011	TASQUEÑA	27/Ago/2009	No enciende el equipo	Repar
2623	18/Sep/2009	TERCERO	26359	2022	GENERAL ANAYA	18/Sep/2009	No enciende el equipo	Repar

Figura 4.6.15 Visualización en pantalla de las fallas atendidas aplicando todos los filtros.



Listado de solicitudes atendidas por fecha

1 de 1 100% Total:2 100% 2 de 2


GERENCIA DE ORGANIZACIÓN Y SISTEMAS
SOPORTE TÉCNICO A P.O.S.
Miércoles, 23 de Septiembre de 2009

Nº de Solicitud	Fecha de Solicitud	Turno	Nº de Taquilla	Estación	Fecha de Atención	Falla	Solución
2,446	27-Ago-09	TERCERO	2011	TASQUEÑA	27-Ago-09	No enciende el equipo	Reparacion Fuente de Poder
2,623	18-Sep-09	TERCERO	2022	GENERAL ANAYA	18-Sep-09	No enciende el equipo	Reparacion Fuente de Poder

Figura 4.6.16 Impresión del reporte de fallas atendidas aplicando todos los filtros.

Modulo Consulta de Pendientes.

En este modulo únicamente se tiene la posibilidad de visualizar todos aquellos reportes a la fecha de consulta se encuentren pendientes de atención, los campos que nos muestra este reporte son: Solicitud, Fecha, Turno, Taquilla, Estación, Falla, Solución (Vea figura 4.6.17).

Modulo Reporte.

En este modulo podemos imprimir el reporte que muestra el desempeño por técnico, es decir, el número de fallas atendidas por cada uno del personal de soporte técnico durante un período de tiempo, así mismo nos da la opción de imprimir o reimprimir una solicitud ya capturada, basta con introducir el número de la solicitud deseada. (Vea figura 4.6.18, 4.6.19 y 4.6.20).



The screenshot shows a web application interface for the Metro de la Ciudad de México. At the top, there is a header with the Metro logo and the text "Metro de la Ciudad de México". To the right of the header is the "Ciudad de México" logo with the motto "Cívica y Honorable". Below the header, there are four tabs: "CAPTURA DE DATOS", "CONSULTA DE ATENDIDOS", "CONSULTA DE PENDIENTES" (which is selected), and "REPORTE".

On the left side of the interface, there is a vertical toolbar with five buttons: "Guardar", "Limpiar", "Buscar", "Imprimir", and "Salir".

The main content area displays a table with the following data:

Solicitud	Fecha	Turno	Taquilla	Estación	Fallas	Solución
2467	31/Ago/2009	SEGUNDO		MOCTEZUMA	Falla de la impresor	Reconexión
2506	07/Sep/2009	PRIMERO		9052 MIXIHUCA	Falla de la impresor	Reconexión
2542	09/Sep/2009	SEGUNDO		5041 OCEANIA	No enciende el equi	Mantenimien
2546	10/Sep/2009	PRIMERO	B061	RIO DE LOS REMED	No enciende el equi	Mantenimien

Figura 4.6.17 Visualización en pantalla de las fallas pendientes.

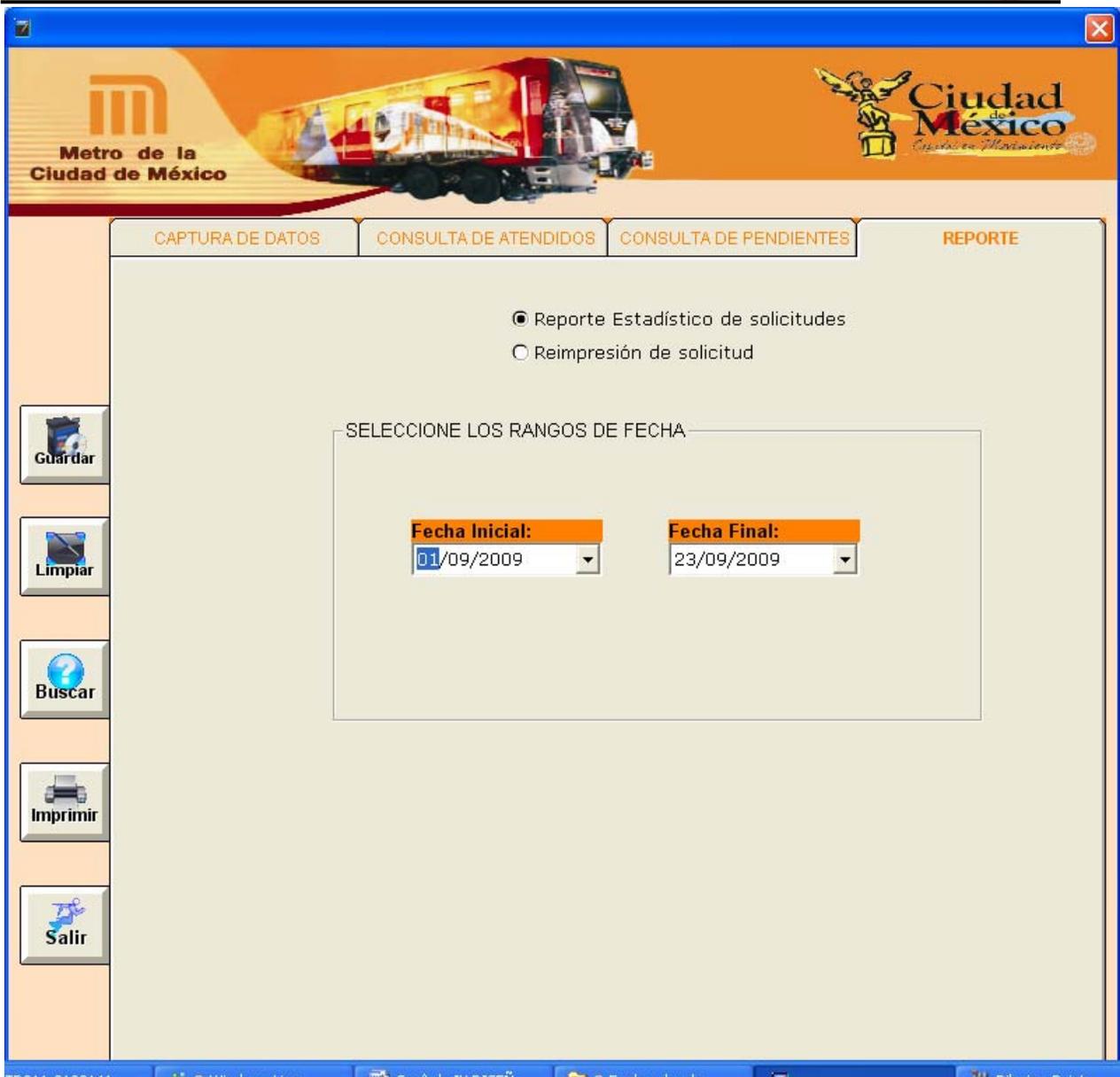


Figura 4.6.18 Selección del reporte estadístico durante el 01 al 23 de septiembre de 2009.



Solicitudes atendidas

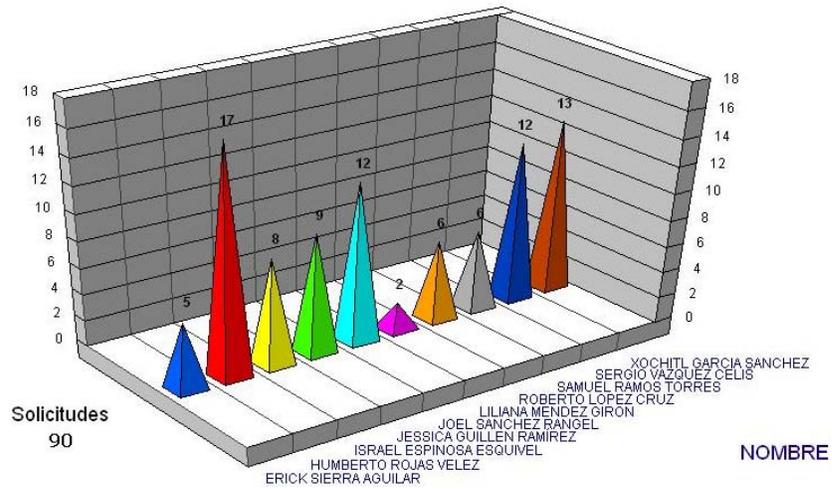


Figura 4.6.19 Impresión del reporte estadístico durante el 01 al 23 de septiembre de 2009.

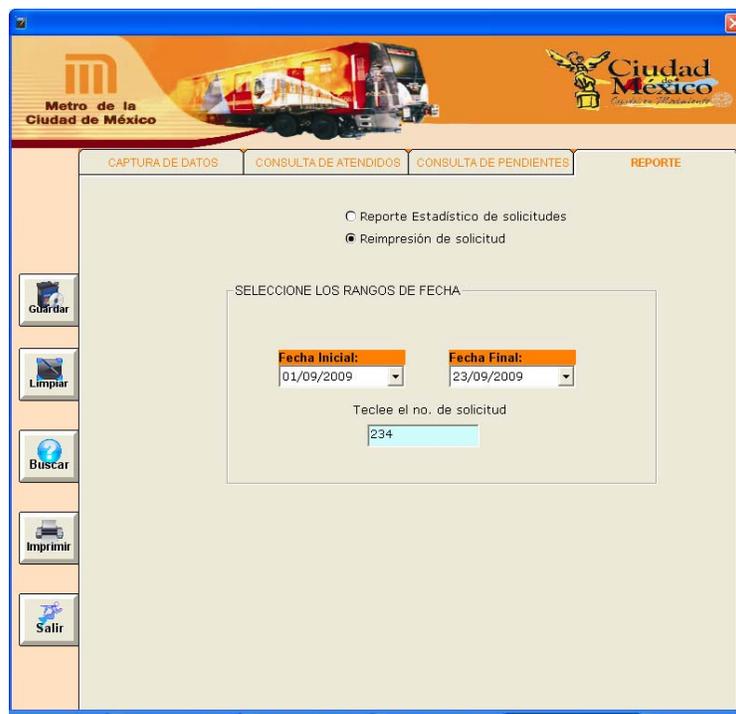


Figura 4.6.20 Reimpresión de Solicitud



CONCLUSIONES.



En la actualidad es ineludible el uso de la tecnología para automatizar tareas que de otro modo son tediosas y altamente susceptibles a errores, además de proveer de un ahorro de tiempo considerable en su ejecución con lo cual se incrementa la productividad y la capacidad de respuesta del área de Soporte Técnico al POS en este caso particular.

Usar una base de datos bien diseñada fortalece tareas de recuperación y almacenamiento de datos logrando así una respuesta favorable del sistema desarrollado y la disposición de información actualizada y confiable en todo momento para su mejor análisis y toma de decisiones de forma rápida y adecuada.

La implementación de una interfaz amigable con el usuario facilita la interacción con nuestro sistema y evita el rechazo hacia el mismo haciendo fácil la operación.

El hecho de desarrollar el sistema dentro de este esquema logró el aprovechamiento tanto de los equipos como del software con que el Sistema de Transporte Colectivo contaba con anterioridad reduciendo costos de manera considerable.

Finalmente el trabajo en equipo permite la conjunción de conocimientos para un desarrollo más rápido que de forma individual esto sin olvidar la importancia de la dirección de nuestro asesor M.I Juan Carlos Roa Beiza para alcanzar nuestro objetivo.



Es de vital importancia al desarrollar un sistema, conocer las necesidades o la problemática que tiene una institución o área específica, para poder brindar una solución con funcionalidad adecuada para dichas necesidades.

Al hacer un análisis final con respecto a los objetivos iniciales de este proyecto, es posible concluir que cada uno de ellos ha sido cubierto con éxito, y que el sistema que se desarrolló puede realizar el registro y control de las fallas de los POS instalados en las taquillas del STC con facilidad, así como la entrega de reportes y la disponibilidad de información precisa en todo momento facilitando la toma de decisiones en cuanto a la adquisición de herramientas y refacciones así como a la asignación y en caso necesario la capacitación del personal técnico en función de la mayor incidencia de fallas.

La mecánica bajo la cual opera el programa de apoyo a la titulación (PAT), resulto para nuestro caso práctica y efectiva ya que nos permitió fijar objetivos desde el principio, dividir la carga de trabajo y conjuntar al mismo tiempo conocimientos prácticos adquiridos en nuestras áreas de trabajo obteniendo al final un sistema completo, práctico y funcional que cubre las necesidades particulares para las cuales fue diseñado.



BIBLIOGRAFÍA.



Administración de los sistemas de información: Organización y tecnología.

Laudon Kenneth C. y Laudon Jane Price.

3ra Edición. Ed. Prentice-Hall, 1996.

Análisis y diseño de Sistemas de Información.

Witten Jeffrey L., Bentley Lonnie D. y Barlow Victor M.

Tercera edición.

Ed. Mc Graw Hill, 1996.

Aprende Microsoft SQL Server 2000 ya.

Rebecca M. Riordan.

Mc Graw-Hill.

Aprendiendo Visual Basic 6 en 21 días.

PERRY GREG.

Prentice Hall, Pearson.

Diseño de interfaces de usuario: estrategias para una interacción persona-computadora efectiva.

Shneiderman, Ben.

Ed Pearson Educación Madrid 2006.

Fundamentos de Bases de Datos.

Silberschatz Abraham, Korth Henry F. y Sudarshan S.

3ra Edición. Ed. Mc Graw Hill, 1998.

Ingeniería de Software un Enfoque Práctico.

Roger S. Pressman.

Ed. Mc Graw-Hill.

México 2006 4ª edición.



Introducción a los Sistemas de Bases de Datos.

Addison Wesley.

Ed. Iberoamericana.

La Biblia de SQL Server 2000.

Patrick Dalton, Paul Whitehead.

Madrid : Anaya Multimedia, 2001.

Sistema de control y registro de información.

Juana Norma Torres Flores.

UNAM, Facultad de Ingeniería, 2006.

Sistemas de Información Gerencial.

Laudon, Kenneth C.

Ed Pearson Educación México 2002.

SQL Server 2000 Programming Step by Step.

RIORDAN M REBECCA.

Microsoft.

URSL's

<http://dgbiblio.unam.mx/tesiunam.html>

<http://microsoft.com/spanish/msdn/articulos/archivo/130701/voices/intfeaturesInSQLserver2000.asp>

http://www.solomantenimiento.com/m_preventivo.htm

<http://www.canalvisualbasic.net/db/tema2.asp>

<http://web.madritel.es/personales3/edcollado/ingsw/tema2/2-1.htm>

<http://www.canalvisualbasic.net/db/tema1.asp>

http://www.itistmo.edu.mx/Pag%20Informatica/APUNTES_archivos/page0006.htm



http://es.wikipedia.org/wiki/Diccionario_de_datos

<http://www.abcdatos.com/tutoriales/tutorial/z5017.html>

http://es.wikipedia.org/wiki/Base_de_datos

http://es.wikipedia.org/wiki/Visual_Basic

<http://www.isatid.net/mm/articulos/sql/sqldbt.htm>

<http://andresorellana.tripod.com/tema2.htm>

<http://www.monografias.com/trabajos10/visual/visual.shtml>

<http://www.auladirectiva.com/curso/programacion-con-visual-basic-6/demostracion-del-curso/content/content1.html>

http://es.wikipedia.org/wiki/Diagrama_de_bloques_de_modelo_matem%C3%A1tico

<http://www.esi2.us.es/~vargas/docencia/tca/ejercicios/ejerciciosCont2.pdf>