



Universidad Nacional Autónoma de México

Facultad de Ingeniería

**SISTEMA INTEGRAL DE CONTROL
DE REPORTES PARA CELULARES**

T E S I S

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

PRESENTAN

DANIEL	ALMAZÁN	CABALLERO
JOSÉ ANTONIO	CORTÉS	CAMACHO
LETICIA	MARTÍNEZ	HERNÁNDEZ
LEONARDO	MIRANDA	ESCAMILLA
OSCAR	TORRES	PIÑA

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



CIUDAD UNIVERSITARIA

Mayo 2008.



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.

A Dios, gracias por todo aquello que me has dado, por darme la oportunidad de finalizar esta etapa, por fortalecerme a cada instante de mi vida y por inundarme de tu presencia.

A mis padres, gracias por la mejor herencia que me han legado, por su paciencia y amor, he aquí el fruto que tanto esperaron.

A todos aquellos que de alguna forma estuvieron involucrados en el desarrollo de la tesis ya que me apoyaron y esperaron a que esta etapa en mi vida finalizara

José Antonio

Agradezco primeramente a Dios por ser mi mejor amigo, mi fortaleza, darme todo lo que tengo y no dejarme caer nunca.

Al M.I Juan Carlos Roa Beiza por asesorarme a lo largo de la tesis y acompañarme en este camino que hoy culmina en el presente proyecto, por compartir su conocimiento.

A mi mamá y mi papá por ser los mejores y estar conmigo incondicionalmente, gracias porque sin ellos y sus enseñanzas no estaría aquí ni sería quien soy ahora, a ellos les dedico esta tesis.

A Miguel, Mario, Laura y Kary por ser la mejor familia que me pudo haber tocado y ser mis hermanos, gracias por el apoyo y confianza que siempre me han brindado.

A mis lindos sobrinos que con sus sonrisas y ocurrencias me hacen ver lo mejor de la vida.

A mis amigos porque gracias a ellos sé lo que es la amistad verdadera, valor importante en mi vida, gracias por estar conmigo, por aconsejarme, regañarme, compartir risas y llantos en todo este tiempo.

Y a Ti por haber aparecido y cambiado mi vida.

Gracias a todos!!

Gracias por ayudarme a lograrlo.

Los quiero mucho

Leticia

A mi esposa;

Gracias por tu amor incondicional, por tus atenciones y detalles, eres de las personas que todo lo comprende y da lo mejor de si, sin esperar nada a cambio. Hoy a tú lado estoy cumpliendo una de mis metas más importantes, gracias por hacerme tan feliz.

Te Amo.

A mis Padres;

Porque gracias a su cariño, guía y apoyo he llegado a realizar uno de mis anhelos más grandes de mi vida, fruto del inmenso amor y confianza que en mi se depositó y con los cuales he logrado terminar mis estudios profesionales, que constituyen el legado más grande que pudiera recibir y por lo cual les viviré eternamente agradecido.

Los quiero mucho.

A mis Hermanas;

Gracias por el apoyo moral y estímulos brindados con cariño y comprensión, porque con su ayuda me alentaron a lograr la terminación de mis estudios, en ustedes encontré a las mejores amigas y siempre podrán contar con mi eterno cariño.

Las quiero mucho.

Leonardo

A mi esposa Alma Luz:

Por su amor y apoyo incondicional en todo momento y ser el impulso para poder concluir esta etapa pendiente de mi vida.

A mis hijos Luz Paula e Isaac:

Por darme las fuerzas y la motivación para ser mejor cada día.

Oscar



ÍNDICE TEMÁTICO

	Página
Capítulo I	
Planteamiento del Problema	
I.1 Introducción	2
I.2 Conceptos fundamentales Empresa y Negocio	6
I.3 Proceso actual del registro de la información	11
I.4 Proceso actual del manejo y organización de la información	15
I.5 Costo tiempo-beneficio de las operaciones	18
Capítulo II	
Fundamentación Teórica	
II.1 Metodologías del diseño y desarrollo de bases de datos	25
II.2 Conceptos de bases de datos y modelo entidad-relación	39
II.3 Características, ventajas y desventajas de SQL Server 2000	45
II.4 Características, ventajas y desventajas de Visual Basic 6	51
II.5 Conceptos del modelo cliente-servidor y de redes	56
Capítulo III	
Análisis y Planteamiento del Problema	
III.1 Diagrama general del proceso e identificación de su problemática	69
III.2 Integración de requerimientos generales y particulares por cada módulo	74
III.3 Elaboración de un diagrama de bloques y sugerencias de posibles soluciones	78
III.4 Análisis de la información recabada para la generación del posible diccionario de datos.	82
III.5 Justificación de la solución a desarrollar para el back-end y front-end	85



Capítulo IV

Análisis, Diseño y Construcción del Sistema

IV.1 Justificación de metodología a utilizar	92
IV.2 Análisis y diseño del back-end	99
IV.2.1 Diagrama de contexto	100
IV.2.2 Diagrama de flujo de procesos	101
IV.2.3 Diagrama de flujo de datos	111
IV.2.4 Diccionario de datos	117
IV.2.5 Diagrama entidad-relación	132
IV.2.6 Normalización de la base de datos	135
IV.2.7 Construcción física de la base de datos	141
IV.3 Análisis, diseño y construcción del front-end	161
IV.3.1 Diagrama jerárquico de funciones del sistema	161
IV.3.2 Desarrollo prueba e integración de las diferentes pantallas con la base	164
IV.4 Pruebas, integración y afinación del sistema	201
IV.5 Mantenimiento y liberación del sistema	212
IV.6 Generación de reportes	230
CONCLUSIONES	236
BIBLIOGRAFÍA	239
MANUAL TÉCNICO Y DE USUARIO	244



Capítulo I



Planteamiento del Problema

I.1 Introducción

El presente trabajo de Tesis se desarrolla para cubrir objetivos como son el llevar a la práctica los conocimientos adquiridos en la Facultad de Ingeniería en la carrera de Ingeniero en Computación.

Desarrollar un sistema para facilitar el control de servicios para celulares, con lo cual se podrá obtener un sistema que gestione toda la información técnica y administrativa que este proceso requiere.

Este diseño es una solución al problema que nos enfrentamos, ya que actualmente los sistemas comerciales que hay en el Mercado Nacional están enfocados a cuestiones administrativas pero no hay ninguno que cubra los aspectos técnicos que un sistema de este tipo requiere. Por tal motivo es necesario desarrollar un sistema hecho a la medida para cubrir las necesidades en todos los aspectos que el cliente desea tener registrados y para ello se sugiere que se utilicen las herramientas de las cuales se disponen las licencias respectivas para evitar problemas legales de derechos de autor y mayores inversiones de software.

El problema básico consiste en que un celular al ser comprado debe ser dado de alta en la base de datos del prestador del servicio y éste genera mucha información técnica que deberá ser recopilada por nuestro sistema para poder comprobar que el teléfono fue adquirido legalmente con un distribuidor autorizado; ya que en caso contrario no se le podrá dar servicio a esos teléfonos.



El sistema deberá incluir los datos de los siguientes procesos:

- Recepción
- Análisis
- Diagnóstico del servicio y la entrega final del reporte de servicio
- Elaboración de comprobantes para la empresa y el cliente

Estos datos nos ayudarán a llevar un registro del histórico de cada equipo recibido, reparado y puesto otra vez en servicio, así como de los irreparables. Diseñar e implantar un sistema que nos permita levantar toda la información referente a los celulares que por alguna causa sufran una falla durante el tiempo de garantía o después de ésta y que deben ser canalizados a un distribuidor que les dará el servicio respectivo.

En la captura se le pedirá al usuario que indique la falla y presente su factura de tal forma que se haga válida la garantía o bien en su caso, en ese momento se le asigne un presupuesto tentativo, el cual será confirmado cuando se realice el proceso de diagnóstico por el laboratorio.

Con éste sistema se pretende dar seguimiento al celular en cuestión desde que ingresa hasta que es reparado o bien en un caso extremo que es diagnosticado como irreparable, también se emitirá un comprobante para el usuario y otro para la compañía que respaldará el servicio prestado al usuario y con el cual se compromete a garantizar el trabajo que recibió su equipo.

El sistema deberá contar con una base de datos relacional lo suficientemente poderosa que nos permita manejar la gran cantidad de información que un sistema de éste tipo requiere; se pretende que tenga un ambiente visual donde de forma intuitiva el usuario pueda navegar de forma ágil y segura por cada uno de los menús.



Lo que se pretende lograr con un sistema intuitivo para el usuario es que éste ingrese la menor cantidad de información ya que ésta deberá estar contenida en catálogos.

Toda la información de los catálogos es información que será actualizada constantemente, para que el sistema contenga toda la información necesaria para el usuario.

Con todos los registros y reportes que se generarán se podrá llevar un histórico de las marcas más confiables en el mercado, esta información beneficiará a la compañía ya que podrá analizar sus costos de servicio para ciertos modelos y marcas en específico.

El objetivo del sistema es facilitar los procesos que se tienen para solicitar el servicio para un celular que ha presentado alguna falla, así como generar reportes y comprobantes para la empresa y el usuario.

Se espera que la solución generada, proporcione beneficios tangibles, para la empresa y los usuarios, facilitando la administración y la operación de la misma, agilizando y potenciando la toma de decisiones, así como los tiempos de respuesta.

El trabajo ha sido estructurado de la siguiente forma:

Planteamiento del Problema

En este capítulo se expondrá el problema actual, ¿cómo se lleva a cabo el registro de la información?, ¿cuánto tiempo toman los procesos?, el actual manejo y organización de la información. Principalmente se definirá el entorno al que nos



enfrentamos, será un bosquejo de la problemática que nos obligó a resolver el problema.

Fundamentación Teórica

Se presentará todas las bases teóricas para fundamentar nuestro sistema, como son las metodologías para el diseño de bases de datos, así como todos los conceptos básicos para el desarrollo. También se proporcionará información del software utilizado para la realización, sus ventajas y desventajas.

Análisis y Planteamiento del Problema

Se analizará cada proceso para identificar su problemática, así como identificar los requerimientos básicos de cada uno con lo cual se podrá plantear las posibles soluciones para cada módulo cubriendo los requisitos de cada proceso, logrando mejoras importantes en tiempo, eficacia, manejo de la información, actualización de catálogos, control de reportes y fácil manejo para el usuario final, ya que éste ingresará pocos datos en el sistema y habrá un control máximo al estar automatizado.

Análisis, Diseño y Construcción del Sistema

Se llevará acabo el análisis de la problemática y se planteará el diseño de la solución, se desarrollaran los diagramas pertinentes para cada proceso y la implementación de éstos. La construcción del sistema se realizará en base a los diagramas de procesos que cubren las necesidades de cada etapa.

Se construirá el sistema y se realizarán las pruebas, integración y afinación de algunos errores para corrección. Así como plantear el mantenimiento del sistema para mejoras continuas.



I.2 Conceptos fundamentales de Empresa y Negocio.

Es innegable que el avance económico y tecnológico de cualquier país esta íntimamente ligado al de su sector industrial, y aunque la administración es aplicable a cualquier grupo social (ya sea educativo, deportivo, militar, etc.), su campo de acción más importante es la empresa.

Primeramente podemos definir **Negocio** como la consecuencia de la correcta administración de los recursos con un resultado económicamente positivo para las partes. Específicamente, negocio puede referirse a entidades individuales de la economía.

La empresa nació para atender las necesidades de la sociedad creando satisfactores a cambio de una retribución que compensara el riesgo, los esfuerzos y las inversiones de los empresarios.

Una **Empresa** es el ejercicio profesional de una actividad económica planificada, con la finalidad o el objetivo de obtener beneficios intermediando en el mercado de bienes o servicios mediante la utilización de factores productivos (trabajo, tierra y capital) y con una unidad económica organizada en la cual ejerce su actividad profesional el empresario por sí mismo o por medio de sus representantes.

La empresa es la unidad económico-social en la que el capital, el trabajo y la dirección se coordinan para realizar una producción socialmente útil, de acuerdo con las exigencias del bien común. Los elementos necesarios para formar una empresa son: capital, trabajo y recursos materiales.

En economía, la empresa es la unidad económica básica encargada de satisfacer las necesidades del mercado mediante la utilización de recursos materiales y



humanos. Se encarga, por tanto, de la organización de los factores de producción, capital y trabajo.

Se entiende por empresa al organismo social integrado por elementos humanos, técnicos y materiales cuyo objetivo natural y principal es la obtención de utilidades, o bien, la prestación de servicios a la comunidad, coordinados por un administrador que toma decisiones en forma oportuna para la consecución de los objetivos para los que fueron creadas. Para cumplir con **este** objetivo la empresa combina naturaleza y capital.

Por último en administración y con base en el análisis de las definiciones anteriores, la empresa es un grupo social en el que, a través de la administración del capital y el trabajo, se producen bienes y/o servicios tendientes a la satisfacción de las necesidades de la comunidad.

Clasificación de las Empresas.

El avance tecnológico y económico ha originado la existencia de una gran diversidad de empresas. Existen numerosas diferencias entre unas empresas y otras. Sin embargo, según en qué aspecto nos fijemos, podemos clasificarlas de varias formas.

Según la actividad económica que desarrolla:

Industriales. La actividad primordial de este tipo de empresas es la producción de bienes mediante la transformación y/o extracción de materias primas. Las industrias, a su vez, son susceptibles de clasificarse en:

- *Extractivas.* Cuando se dedican a la explotación de recursos naturales, ya sean renovables o no renovables, entendiéndose por recursos



naturales todas las cosas de la naturaleza que son indispensables para la subsistencia del hombre.

- *Manufactureras.* Son empresas que transforman las materias primas en productos terminados, y pueden ser de dos tipos:
 - *Empresas que producen bienes de consumo final,* producen bienes que satisfacen directamente la necesidad del consumidor.
 - *Empresas que producen bienes de producción,* éstas satisfacen preferentemente la demanda de las industrias de bienes de consumo final.

Comerciales. Son intermediarias entre productor y consumidor; su función primordial es la compra-venta de productos terminados.

Pueden clasificarse en:

- *Mayoristas:* Cuando efectúan ventas en gran escala a otras empresas (minoristas), que a su vez distribuyen el producto directamente al consumidor.
- *Minoristas o detallistas.* Las que venden productos al menudeo, o en pequeñas cantidades, al consumidor.
- *Comisionistas:* Se dedican a vender mercancía que los productores les dan a consignación, percibiendo por esta función una ganancia o comisión.

Servicio. Como su denominación lo indica, son aquellas que brindan un servicio a la comunidad y pueden tener o no fines lucrativos.



Las empresas de servicio pueden clasificarse en:

- Transporte.
- Turismo.
- Instituciones Financieras.
- Servicios públicos varios: Comunicaciones, energía, agua.
- Servicios privados varios: Asesoría, diversos servicios contables, jurídicos, administrativos, promoción, ventas y agencias de publicidad.
- Educación.
- Salubridad (hospitales).
- Finanzas.

Las empresas también se pueden clasificar según su dimensión. No hay unanimidad entre los economistas a la hora de establecer qué es una empresa grande o pequeña, puesto que no existe un criterio único para medir el tamaño de la empresa. Los principales indicadores son: el volumen de ventas, el capital propio, número de trabajadores, beneficios, etc. El más utilizado suele ser según el número de trabajadores. Este criterio delimita la magnitud de las empresas de esta forma:

- *Microempresa*: si posee menos de 10 trabajadores.
- *Pequeña empresa*: si tiene menos de 50 trabajadores.
- *Mediana empresa*: si tiene un número entre 50 y 250 trabajadores.
- *Gran empresa*: si posee más de 250 trabajadores.

Características de una Empresa.

Una empresa combina tres factores que son:



- *Factores activos:* empleados, propietarios, sindicatos, bancos, etc.
- *Factores pasivos:* materias primas, transporte, tecnología, conocimiento, contratos financieros.
- *Organización:* coordinación y orden entre todos los factores y las áreas.

Factores Activos.

Personas físicas y/o jurídicas (otras entidades mercantiles, Cooperativa, fundaciones, etc.) constituyen una empresa realizando, entre otras, aportación de capital (sea puramente dinerario, sea de tipo intelectual, patentes, etc.). Estas "personas" se convierten en accionistas de la empresa.

- Participan, en sentido amplio, en el desarrollo de la empresa:
- Administradores
- Clientes
- Colaboradores
- Fuentes Financieras
- Accionistas
- Suministradores y proveedores
- Trabajadores.

Factores Pasivos.

Todos los que son usados por los elementos activos y ayudan a conseguir los objetivos de la empresa. Como la tecnología, las materias primas utilizadas, los contratos financieros de los que dispone, etc.

Dentro de una empresa hay varios departamentos, o *áreas funcionales*. Una posible división es:



- Producción y logística
- Dirección y Recursos Humanos
- Comercial(Marketing)
- Finanzas y Administración
- Sistemas de Información
- Ventas

Finalmente, de acuerdo a la anterior definición y clasificación de empresa, así como, al estudio realizado para la determinación del área correspondiente de la aplicación del Sistema Integral de Control de Reportes para Celulares, se ha dictaminado que está diseñado para aquellos establecimientos que se encuentran en la sección de servicios privados varios, cuya principal finalidad es la prestación de servicios en reparación de aparatos de telefonía celular, ya que cubre con las principales necesidades para poder ofrecer un servicio de calidad, de manera rápida y eficiente.

I.3 Proceso Actual del Registro de la Información

En este capítulo describiremos la forma como se realiza el registro de la información cuando se reporta una falla en los equipos celulares hasta la entrega del equipo al cliente. El proceso del registro de la información comienza cuando el cliente reporta la falla de su teléfono celular con el operador del centro de atención, en este caso se realiza el llenado del “Formato Recepción de Equipo” (Fig. I.3.1) en donde se solicitarán los datos más importantes del usuario, entre ellos: nombre, domicilio y teléfono por mencionar algunos, también dentro de este formato se agregan los datos del aparato telefónico, así como la falla que reporta el usuario y el costo, cotización o garantía de la reparación del equipo. Este formato se acumula con las demás solicitudes que son recabadas durante el día para su atención.



FORMATO RECEPCIÓN DE EQUIPO

ORDEN DE SERVICIO	FOLIO DE GARANTIA	FECHA	HORA
1.- DATOS CLIENTE			
NOMBRE(S) Y APELLIDO _____			
R.F.C _____		EMAIL _____	
EMPRESA _____			
DOMICILIO _____			
C.P. _____		MUNICIPIO / DELEGACION _____	ESTADO _____
TELEFONO _____			
2.- DATOS DEL EQUIPO			
MODELO _____		TECNOLOGIA _____	
IMEI _____	MSN _____	ESN/CSN _____	
COD. FABRICACION _____		VER. SOFTWARE _____	
TARJETA SIM _____		COD. CARRIER _____	
3.- FALLA			
GOMA	<input type="checkbox"/>	TAPA	<input type="checkbox"/>
CARGADOR	<input type="checkbox"/>	HOTLER	<input type="checkbox"/>
BATERIA	<input type="checkbox"/>	TAPON	<input type="checkbox"/>
ANTENA	<input type="checkbox"/>	MANOS LIBRES	<input type="checkbox"/>
INTERFAZ	<input type="checkbox"/>	BLUETHOOTH	<input type="checkbox"/>
MEMORIA	<input type="checkbox"/>	OTRO	_____
4.- GARANTIA / COTIZACION			
FECHA DE FACTURA _____		COTIZACION _____	
GARANTIA POR FACTURA _____		COSTO _____	
_____		_____	
ENTREGA		RECIBE	

Fig. I.3.1 Formato Recepción de Equipo

En el área del laboratorio se tiene otro formato en el cual se registra la información más importante de los equipos celulares, así como las fallas encontradas dentro de los aparatos y las reparaciones que se hicieron a éstos. También se registra el nombre del técnico que atendió la solicitud y el tiempo de solución. La Fig. I.3.2 muestra el “Formato Laboratorio”.

**FORMATO LABORATORIO**

ORDEN DE SERVICIO	FOLIO DE GARANTIA	FECHA	HORA
1.- DATOS DEL EQUIPO			
MODELO _____	TECNOLOGIA _____		
IMEI _____	MSN _____	ESN/CSN _____	
COD. FABRICACION _____	VER. SOFTWARE _____		
TARJETA SIM _____	COD. CARRIER _____		
GARANTIA _____	COTIZACION _____	COSTO _____	
2.- DATOS TÉCNICOS			
TÉCNICO _____	ID TÉCNICO _____		
SALIDA PROGRAMADA _____	FECHA _____	HORA _____	
	TIEMPO DE REPARACIÓN _____		
FALLAS ENCONTRADAS _____			
SOLUCIÓN DE FALLAS _____			
OBSERVACIONES _____			

Fig. I.3.2 Formato Laboratorio

Finalmente cuando el equipo es devuelto del centro de atención al cliente, se informa al usuario que puede pasar por su equipo telefónico en la fecha indicada. Una vez que el cliente acude a recoger el equipo, se llenará el formato de entrega de equipo, en el cual se pedirán nuevamente los datos del cliente, se ingresaran los datos del equipo, la solución y el costo de reparación o la aplicación de garantía según sea el caso. La Fig. I.3.3 muestra el “Formato Entrega de Equipo”.

**FORMATO ENTREGA DE EQUIPO**

ORDEN DE SERVICIO	FOLIO DE GARANTIA	FECHA	HORA
1.- DATOS CLIENTE			
NOMBRE(S) Y APELLIDO _____			
R.F.C _____		EMAIL _____	
EMPRESA _____			
DOMICILIO _____			
C.P. _____		MUNICIPIO / DELEGACION _____	ESTADO _____
TELEFONO _____			
2.- DATOS DEL EQUIPO			
MODELO _____		TECNOLOGIA _____	
IMEI _____	MSN _____	ESN/CSN _____	
3.- SOLUCION			
GOMA	<input type="checkbox"/>	TAPA	<input type="checkbox"/>
CARGADOR	<input type="checkbox"/>	HOTLER	<input type="checkbox"/>
BATERIA	<input type="checkbox"/>	TAPON	<input type="checkbox"/>
ANTENA	<input type="checkbox"/>	MANOS LIBRES	<input type="checkbox"/>
INTERFAZ	<input type="checkbox"/>	BLUETHOOTH	<input type="checkbox"/>
MEMORIA	<input type="checkbox"/>	OTRO	_____
FECHA Y HORA DE ENTREGA _____			
OBSERVACIONES _____			
4.- COSTO DE REPARACION			
COSTO _____		FORMA DE PAGO _____	
GARANTIA _____			
_____		_____	
ENTREGA		RECIBE	

Fig. I.3.3 Formato Entrega de Equipo



I.4 Proceso actual del manejo y organización de la información

El sistema que se utiliza para la elaboración de los reportes de servicio no permite su administración de una manera eficaz y adecuada, lo que se traduce en una serie de deficiencias en el manejo de la información recabada. El personal de atención al cliente levanta la orden de servicio o en su defecto toma el folio de garantía en caso que el equipo cuente con ella, así como la fecha y hora en que se realiza la solicitud, llenando el formato de recepción del equipo, en el cual se solicitan los datos del cliente, los datos del equipo y se anotara la falla que presente el equipo, por último se informará al usuario del costo de la reparación o se le proporcionará una cotización.

La información referente a las averías que afectan a los teléfonos celulares pasa por un proceso que se inicia desde el momento en el que se recibe el teléfono que por alguna causa ha sufrido una falla, después se solicita mayor información la cual nos permitirá poder identificar de una manera más sencilla ¿cuál es la posible causa de la avería?.

En esta etapa se requiere que el usuario aporte una descripción muy completa y detallada de las razones por las cuales el teléfono presenta esa (s) falla(s), el tiempo desde que se inicio la problemática, etc.

De esta manera no se puede llevar un control detallado del número de reportes que se van generando en el transcurso del día.

Una vez que se tiene una descripción general del tipo de falla que presenta el teléfono celular se procede a elaborar un reporte para el análisis del mismo, dicho análisis es realizado por un técnico especializado que se encarga de determinar de una manera precisa ¿cuál es el problema que afecta al equipo? y la posible solución del mismo.



El diagnóstico elaborado por el técnico es notificado al cliente, el cual una vez al tanto de la situación que presenta su equipo celular procede a informar a la recepción si esta de acuerdo con la solución propuesta por el técnico, en caso de que el cliente esté de acuerdo con la propuesta de reparación se envía el teléfono celular al centro de servicio respectivo para su reparación.

Solucionada la falla el técnico llena el formato de laboratorio en el cual se pide describa las fallas que se encontraron, la solución a las fallas, la salida programada del equipo y el nombre del técnico que revisó el equipo, para que sea devuelto al centro de atención a clientes.

Los equipos se entregan en el centro de atención a clientes, con el formato de laboratorio para su registro.

Se le informa al usuario de la reparación del equipo, notificándole la fecha y hora de su entrega.

Se realiza el llenado del formato de entrega de equipo donde se indica el nombre del cliente, los datos del equipo, la solución de la falla y el costo de la reparación o la solución debido a la garantía del equipo.

De lo contrario, el teléfono celular es remitido nuevamente a la recepción para que sea entregado al cliente. Ver figura 1.4.1.

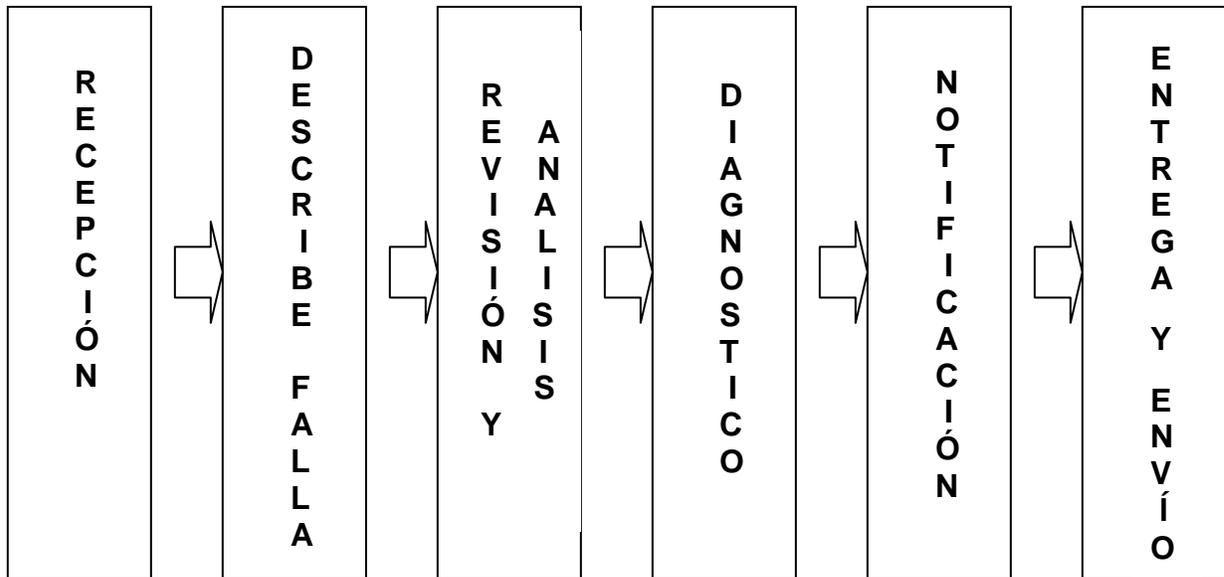


Figura I.4.1 Diagrama de bloques del flujo de la información.

Problemática del flujo de información

Debido a que actualmente el registro de la información de los clientes y sus equipos celulares se realiza en bitácoras y hojas de servicio que son capturadas manualmente o en hojas de cálculo por el responsable de esta función y el cual no lleva un control eficaz para después recuperar dicha información con la premura que se requiere en cada caso, el problema del flujo de información consiste en que al adquirir un teléfono celular es dado de alta en la base de datos del prestador del servicio, el cual genera mucha información técnica (marca, modelo, etc.), la cual deberá ser recopilada por el sistema a desarrollar, con lo cual se comprobará que el teléfono fue adquirido legalmente con un distribuidor autorizado, de lo contrario no se le prestará el servicio al dueño de dicho equipo.



Otro problema es que al no contarse con catálogos que describan las fallas más frecuentes que se presentan en los teléfonos celulares, se requiere de una mayor cantidad de tiempo para poder determinar la causa posible que ocasiona la falla, lo cual se traduce en un periodo mayor de tiempo requerido.

El proceso de entrega-recepción es ineficaz y se demora mucho por que no se cuenta con un sistema de reporte que permita identificar a cada equipo recibido a través de un número único, el cual nos permitiría saber de una manera rápida, sencilla y eficaz el estatus de cada uno de los teléfonos a revisar.

Dado que el sistema con el que se cuenta esta diseñado para satisfacer necesidades generales, no permite expedir comprobantes de servicio tanto para el cliente como para el usuario.

No se cuenta con una base de datos propia que nos permita llevar un archivo histórico (bitácora) de cada equipo recibido, de aquellos que han sido reparados y están en servicio nuevamente, así como de los irreparables, ya que se tiene que recurrir a la información que proporcionan terceros y que en algunas ocasiones dista de ser precisa. Por lo anterior no es posible darle certidumbre al usuario de que el equipo que esta comprando posee el respaldo de una compañía.

I.5 Costo tiempo – beneficio de las operaciones

Para implementar el Sistema Integral de Control de Reportes para Celulares se deben examinar, e identificar asimismo claramente los objetivos de la aplicación de este sistema. Las preguntas que se deben tomar en cuenta incluyen: los beneficios de la introducción del sistema, la eficiencia y la efectividad que se tiene



actualmente y la que se tendrá con la implementación, así como si los beneficios que se obtendrán serán mayores a los costos.

La forma en la que actualmente se registra la información y así como el manejo y organización de la misma causa una mala gestión del tiempo y pérdida de recursos debido a que existen factores que perjudican la eficiencia del personal, como es, el tiempo que se dedica al alta de nuevos clientes dentro de la compañía representa pérdidas de tiempo, esto ocurre como una consecuencia del proceso consistente en capturar los datos de los nuevos clientes en una hoja de Excel, ya que un cliente fue dado de alta inicialmente y la búsqueda de cada uno de las personas es de forma manual, y en ocasiones esto ocasiona redundancia de datos.

Actualmente la información que se maneja nos obliga a buscar los datos del cliente de forma manual, ocasionando con esto que la actualización de la misma o del celular nos origine un retraso, con el nuevo sistema se pretenderá agilizar el mencionado proceso, ya que cada uno de los clientes contará con un número que lo identifique de forma inmediata, o en dado caso de no contar con ese número se podrá buscar por su nombre reduciendo con ello el tiempo de búsqueda y actualización de los datos.

Otra de las pérdidas de tiempo que existe se origina cuando se da de alta un celular en la base de datos y éste pasa a la etapa de reparación, ya que se captura cada uno de ellos al momento de entrar a la compañía y no se tiene una historia de cuáles han sido aquellos celulares que anteriormente se han captado en forma inmediata, y las fallas que han presentado, originando con esto que cada vez que un celular entra al laboratorio se debe de diagnosticar nuevamente sin tener un antecedente.



El costo que la compañía maneja en sus operaciones se refleja en los siguientes elementos: costo de la jornada de un trabajador, tiempo que tarda un trabajador con cada cliente, seguimiento de equipo durante el proceso de reparación, materiales para la emisión de reportes.

Cada uno de los trabajadores obtiene una remuneración la cual es una consecuencia del número de horas que trabaja, se observa que con el actual sistema de trabajo, éste refleja pérdidas de tiempo en la búsqueda de información de los celulares, del estado de reparación y de los clientes. Esto ocasiona pérdidas económicas ya que un empleado tiene un promedio de 12 minutos al dar de alta un cliente y el celular en cuestión, además de los tiempos agregados si el cliente ya existía anteriormente en el sistema. Con el nuevo sistema se pretenderá disminuir el número de minutos en la captura de la información del cliente y del celular.

El tiempo de respuesta actual para una consulta del estado de reparación de un celular depende directamente del trabajador que atiende al cliente y de los trabajadores del laboratorio, ocasionando con esto perdidas de tiempo, con la implementación del sistema se pretende obtener:

- Rápida provisión de la información
- Amplio espectro de información proporcionada
- Mayor precisión de la información brindada

Otra de las dificultades que se presentan, se debe a la tardanza que existe para tener un seguimiento del estado de reparación de cada uno de los equipos celulares, ya que ésta se lleva de forma manual con la localización de los reportes que se emiten.



A esto se le puede agregar la pérdida de material en la emisión de reportes, y en el laboratorio, ya que al no llevarse un histórico del equipo que ha estado en reparación con anterioridad, se tiene que evaluar varias veces un mismo equipo ocasionando con esto que el personal adscrito al laboratorio realice varias veces un mismo diagnóstico y como consecuencia la misma reparación en varias ocasiones con su consecuente gasto de piezas, con el nuevo sistema se espera reducir el costo de partes de reparación, la disminución de horas extras por parte de los empleados, así como una menor pérdida de material al emitir un reporte varias veces.

Otro de los elementos que dañan la efectividad y eficiencia en el sistema se origina en los catálogos existentes de cada una de las diversas compañías de telefonía celular, en donde se tienen claves de las fallas que presentan los equipos y al momento que un empleado este ingresando un equipo para su diagnóstico se tenga que buscar en una lista la falla existente y su correspondiente clave, lo cual origina una pérdida de tiempo. El sistema que se propone tendrá la posibilidad de importar dichos catálogos, provenientes de cada una de las diversas compañías celulares, y en el momento en que se este dando de alta uno de los equipos se puedan utilizar los catálogos de forma inmediata y rápida, para disminuir así el tiempo de captura y consulta de fallas.

De esta manera se logrará confirmar y entregar un pedido más rápido gracias a la nueva interfaz gráfica que permitirá visualizar un mayor número de datos y la búsqueda de la información de forma más rápida, mejorando así la calidad del servicio.

Una de los beneficios que se presentarán al implementar el nuevo sistema, será hacer un diagnóstico de las marcas y equipos más confiables para los distintos tipos de usuarios existentes en el mercado, ya que al contar con un histórico de



las fallas existentes en cada uno de los celulares, se podrá contar con bases firmes para indicar las fallas recurrentes que se presentan en algunos equipos, ocasionando con esto que la compañía no adquiera un producto propenso a fallas y así tener la menor cantidad de equipos en garantía y utilizar el tiempo de los empleados en la reparación de equipos que generen ingresos por concepto de reparación.

Otro de los beneficios indirectos que se obtendrá será la confiabilidad del cliente en la empresa, en la entrega de equipos que cumplan con los requerimientos que se esperan al momento de su adquisición ya que al llevar el histórico de las marcas y los modelos, se presentarán una menor cantidad de fallas por concepto de garantía dando certidumbre al cliente y un óptimo manejo de la calidad del producto final en el mercado.

Al ser desarrollado el sistema en un ambiente visual, se esperará que la adaptación de los trabajadores sea de forma rápida debido a que la interfaz gráfica será elaborada de forma intuitiva para poder navegar de forma ágil y segura por cada uno de los menús, reduciendo con esto el tiempo que se necesitará preparar a los trabajadores minimizando así el costo de capacitación en la implementación del nuevo sistema.

El sistema contemplará el acceso por medio de una contraseña, la cual tiene como finalidad controlar el acceso a ciertas áreas de acuerdo al nivel que posea cada empleado, logrando con ello el beneficio de acceso a la información pertinente que cada uno de los empleados necesite, evitando pérdidas de información.



Otro de los beneficios del sistema a implementar consiste en la consulta del personal que atendió cada uno de los casos, esto servirá al departamento de recursos humanos para conocer el nivel de respuesta por parte de cada uno de los trabajadores teniendo así un histórico de la eficiencia y eficacia del empleado.

Contemplando todo lo anterior, al poner en marcha el nuevo sistema se disminuirán los costos, se controlará la eficiencia de cada trabajador, logrando un menor tiempo de respuesta al cliente y un estado real del celular que se encuentra en proceso de reparación en el laboratorio, se tendrá por parte del cliente mayor confianza en el trabajo realizado debido al histórico, sustitución del papel por documentos electrónicos y la posibilidad de revisar los datos ingresados tantas veces como se necesite.



Capítulo II



Fundamentación Teórica

II.1 Metodologías del Diseño y Desarrollo de Bases de Datos

2.1.1 Ingeniería de Software

Es la rama de la ingeniería que crea y mantiene las aplicaciones de software con tecnologías y prácticas de las ciencias computacionales, manejo de proyectos, el ámbito de la aplicación, y otros campos. La ingeniería de software, como las disciplinas tradicionales de ingeniería, tiene que ver con el costo y la confiabilidad.

2.1.2 Metodologías

Un objetivo de décadas ha sido el encontrar procesos o metodologías predecibles y repetibles que mejoren la productividad y la calidad.

La ingeniería de software tiene varios modelos o paradigmas de desarrollo en los cuales se puede apoyar para la realización de software, de los cuales podemos destacar a éstos por ser los más utilizados y los más completos:

- Metodología de SSADAM
- Metodología Merisse
- Ciclo de Vida estructurado (Yourdon y De Marco)



2.1.3 Metodología de SSADAM

Por sus siglas en inglés SSADAM (Structured Systems Analysis and Design Method) ésta metodología fue producida por la Agencia Central de Computación y Telecomunicaciones del Reino Unido hacia 1980.

SSADM es una disciplina del desarrollo de Software que busca proveer de un marco de referencia para las actividades de captura, almacenamiento, transformación y diseminación de información habilitados por sistemas computacionales, la cual descansa en el trabajo de varias escuelas de métodos de desarrollo, uno de los aportadores clave es:

- Ed Yourdon, desarrollador de la Metodología de Programación Estructurada.

SSADAM es una metodología de cascada y hace el desarrollo de software más fácil, rápido, efectivo y tiene un enfoque riguroso de documentación.

Etapas

La metodología SSADAM involucra la siguiente secuencia de pasos de análisis, documentación y diseño:

Análisis de Factibilidad.- Analiza la situación actual y su problemática a un alto nivel, utilizando los Diagramas de Flujo de Datos (DFD) para documentar el sistema actual y visualizar los problemas.



Los siguientes pasos son parte de esta etapa:

- Investigación sobre las actividades del negocio, levantamiento de los principales eventos y reglas de negocio.
- Levantamiento de Requerimientos, Identificar los problemas asociados con el entorno actual, servicios adicionales y usuarios que el nuevo sistema debe atender.
- Documentar el proceso actual. Documentando en un Diagrama de Flujo (DF) el servicio actual e identificando sus deficiencias.
- Documentar datos actuales. En éste paso se identifican y describen las estructuras de datos, independientemente de la manera en la que son procesados y organizados actualmente.

Análisis de requerimientos.- La primera parte de ésta etapa corresponde a la investigación del entorno actual en la que se definen los requerimientos y el modelado del negocio creando un DFD y las estructuras de los datos que serán parte del sistema. En segundo lugar se describen las opciones de solución y se selecciona una de ellas para construcción. Los siguientes pasos son parte de ésta etapa:

- Definir las opciones de solución posibles que cubran los requerimientos de negocio
- Seleccionar la solución de negocio. La solución elegida define el alcance del sistema a ser desarrollado en las etapas siguientes.

Especificación de requerimientos.- También conocida como Etapa de especificación detallada de requerimientos de negocio. Los siguientes pasos son parte de ésta etapa:



- Definir y especificar el procesamiento que se requiere automatizar. Describe el sistema en términos de flujo de datos y define los roles de usuario del nuevo sistema.
- Desarrollo del modelo de datos normalizado
- Definir las funciones del sistema. Se describen los eventos y funcionalidades requeridas y se identifican los niveles de servicios solicitados.
- Desarrollar procedimientos de usuario para entender las actividades de los usuarios.
- Desarrollar la especificación de los prototipos.
- Confirmación de los objetivos del sistema, como revisión final de los requerimientos antes de la conclusión de la etapa de definición.

Diseño lógico del sistema.- En ésta etapa se selecciona una opción de solución técnica factible, sobre la que son basados el desarrollo e implementación del sistema. Los siguientes pasos son parte de ésta etapa:

- Definir opciones técnicas de solución. Valida también los niveles de servicio requeridos a la luz de los requerimientos del ambiente técnico.
- Selecciona la Opción Técnica del Sistema en conjunto con el usuario.

Especificación lógica del sistema.- Tomando como base la solución técnica seleccionada, son actualizados los diseños lógicos y los de procesos. Adicionalmente se especifican las interfaces de usuario. Los siguientes pasos son parte de ésta etapa:

- Definir la interfaz de usuario. La estructura de los diálogos requeridos para soportar las funciones en línea del sistema, identificando los requerimientos de navegación.



- Definir actualizaciones a los procesos. Para completar la especificación de la Base de Datos requerida para cada evento y para definir el manejo de errores de cada evento.
- Definir requerimientos de consulta.

Diseño Físico.- El objetivo de ésta etapa es especificar el diseño físico de datos y procesos, usando un lenguaje con características afines al ambiente técnico seleccionado. Las siguientes son actividades de ésta etapa:

- Preparación para el diseño físico.
- Establecer reglas de implementación del ambiente.
- Revisar los requerimientos detallados del mapeo lógico y físico.

Técnicas

Las tres técnicas más importantes de la Metodología SSADAM son:

- **Modelado Lógico de los datos.** Es el proceso de identificar, modelar y documentar los datos requeridos en el sistema en diseño. Los datos son separados en entidades (cosas de las cuales el sistema necesita guardar información) y relaciones (las asociaciones entre entidades).
- **Modelado de Flujo de Datos.** Es el proceso para identificar, modelar y documentar como se mueven los datos en el sistema, examinando los Procesos (actividades que transforman datos de una forma a otra), Almacenes de datos (las áreas de almacenamiento de datos), Entidades externas (que envían datos al sistema o que reciben datos del sistema) y Flujo de datos (rutas por las cuales los datos pueden fluir).
- **Modelado de Procesos.** Se refiere a modelar y documentar los eventos que afectan cada entidad y la secuencia en la que estos eventos ocurren.



2.1.4 Metodología MERISSE

El proyecto Merisse fue comenzado en 1977 por el Ministerio de Industria Francés, es un método de concepción y de desarrollo de sistemas de información. Se concibe el sistema de información como un objeto artificial intermediario entre el sistema que opera y el sistema de conducción. Éste enfoque mejora notablemente la calidad de las especificaciones antes de la realización y sobretodo conduce a sistemas que evolucionan. Abarca los aspectos relacionados con la recopilación y validación de la información, capacitación de personal, valuación de equipos informáticos, análisis, diseño y validación de los procesos e implementación, gestión de costos - tiempos, y el desarrollo del código.

El proceso Merisse se realiza según tres ejes a saber:

- Ciclo de Abstracción
- Ciclo de Vida
- Ciclo de Decisión

En cada etapa del ciclo de vida, se utilizan, con una precisión cada vez mayor, los formalismos del ciclo de abstracción, y se toman decisiones, al principio de forma global, y después de forma más detallada, conforme va progresando en el trabajo.

Los tres ciclos se desarrollan simultáneamente.

Ciclo de Abstracción.- Merisse utiliza tres formalismos, de lo más abstracto a lo más concreto para modelar un sistema de información y se aplican a los procesos y a los datos. Estos tres formalismos corresponden a los tres niveles: Conceptual, Organizacional y Operacional.



Ciclo de Vida.- Incluye tres actividades: Concepción, Realización y Mantenimiento.

Ciclo de decisión.- Un aspecto muy importante de Merisse es que se ocupa al mismo tiempo del estudio de los datos y de los procesos. La jerarquía de decisiones es como se explica a continuación:

- Descomposición del sistema de información en dominios.
- Orientaciones en materia de gestión y organización.
- Planificación de desarrollo
- Elección entre procedimientos manuales y automáticos
- Elección entre procedimientos en tiempo real y automatizados
- Determinación de puestos de trabajo y tareas
- Diseños de estados, pantallas, etc.

Etapas del Ciclo de Vida de un sistema de información

Etapa1: Esquema Director.- Consiste en estudiar el sistema de forma global, descomponiéndolo en dominios. Se fijan las grandes orientaciones y se obtiene un plan de Desarrollo. Se establece un puente entre los objetivos estratégicos de la organización y sus necesidades de información.

Etapa 2: Estudio Previo.- Se realiza para cada dominio a estudiar. Se parte de la situación existente de la que se obtiene el Modelo Organizacional de tratamientos (MOT) y el Modelo Lógico de Datos (MLD) actuales. Se elimina la organización existente y se deduce el Modelo Conceptual de Tratamientos (MCT) y el Modelo Conceptual de Datos (MCD) actuales.



Teniendo en cuenta las consideraciones generales fijadas por la dirección general y las críticas realizadas durante el estudio del sistema existente se modifica el MCT y el MCD para obtener el MCD y el MCT futuros.

Se crean varios escenarios para la nueva organización y se establece para cada uno de ellos el grafo de circulación de los procedimientos más representativos (MOT futuro parcial).

Se define el hardware y software a utilizar y se termina con una evaluación de cada escenario, en términos de costos, ventajas, impactos en la organización, etc.

Etapa 3: Estudio Detallado.- Para cada dominio y a partir del escenario elegido en el estudio previo, se procede a un estudio detallado, realizándose el MOT, validación del MCD y el MLD.

El dominio se puede descomponer en proyectos, realizándose el estudio para éstos. Cada proyecto se descompone en aplicaciones, redactándose un Cuaderno de cargas de usuarios por aplicación.

Etapa 4: Estudio Técnico.- Se compone de los siguientes pasos: Optimización del MLD, MCD y MOT. Da lugar a la realización de un Cuaderno de Cargas de Realización por aplicación.

Etapa 5: Realización.- Comprende por cada aplicación:

- Programación y Pruebas
- Implementación

Etapa 6: Mantenimiento.- Se ajusta el sistema a nuevas necesidades, previstas o no, se corrigen errores detectados en el uso del sistema.



2.1.5 Ciclo de Vida estructurado (Yourdon y De Marco)

Los principales exponentes de esta metodología son: Yourdon y De Marco. El Ciclo de Vida Estructurado (CDVE), surge como una evolución del Ciclo de Vida Clásico (CDVC), en entornos grandes y complejos. En la actualidad se conocen como análisis estructurado, diseño estructurado y programación estructurada, pero este método es bastante más que esto. El CDVE es un ciclo lineal o en cascada, que provee técnicas y herramientas adecuadas para cada fase del proyecto. Es una metodología que ofrece mayores puntos de control para el proyecto. Es más flexible, lo que facilita el mantenimiento (modificación y adaptación del sistema). Junto con esta metodología aparecen nuevos conceptos como la modularización y el diseño descendente. En la siguiente figura 2.1.5.1 se presentan las fases involucradas:

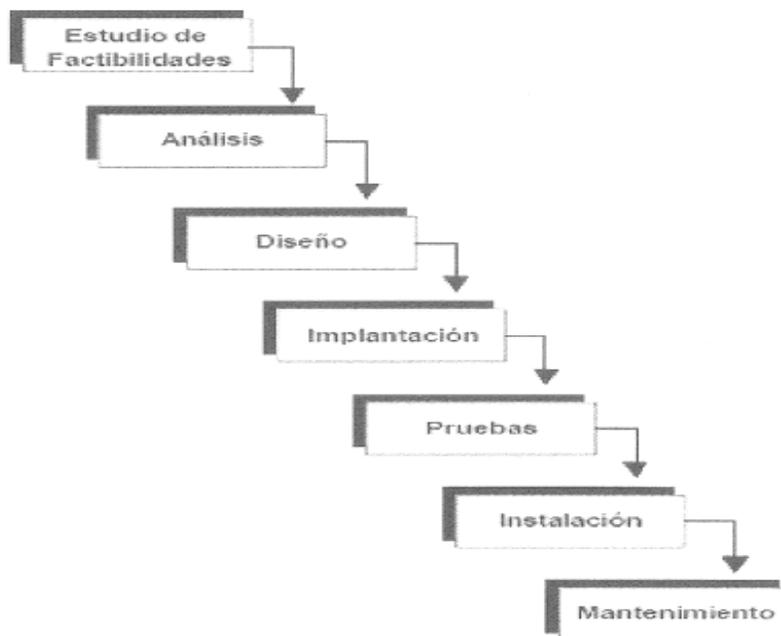


Figura 2.1.5.1 Fases de Metodología Estructurada



Estudio de Factibilidad.- Consiste en una visión general, de alto nivel, de todo el proyecto, destinado a contestar a un número determinado de preguntas: ¿Cuál es el problema? ¿Existe una solución viable? ¿Cuánto tiempo llevará? ¿Cuál es el costo?

El estudio de factibilidad debe ser relativamente corto, pues la tarea no consiste en resolver el problema, sino en obtener una idea de su complejidad y tamaño. Éste comienza con la aclaración de la definición del problema, se definen el alcance y los objetivos iniciales y se identifican las restricciones y limitaciones del sistema; y se preparan varias opciones de solución. Para cada una de ellas se analiza:

- Factibilidad Técnica: ¿Puede el software implementarse empleando la tecnología actual y disponible?
- Factibilidad Económica: ¿Superan los beneficios los costos?
- Factibilidad Operativa: ¿Puede el software desarrollarse y ejecutarse en esta organización?

Análisis.- Ésta fase intenta determinar los requerimientos a partir del estudio del sistema de información existente, a manera de incluir todos los detalles relevantes del sistema actual, además pretende:

- Que sea fácil detectar y verificar la omisión de detalles relevantes
- Que distintos analistas ante el mismo sistema actual determinen los mismos requerimientos
- Que la documentación generada sobre el sistema actual sean vehículos eficientes de comunicación.



El analista se familiariza con la operación del sistema, para ello utiliza una serie de técnicas y herramientas. En ésta fase se obtiene un modelo del sistema existente que describe las funciones del sistema, los datos, el control, etc. El documento resultado de ésta etapa se suele llamar *especificación estructurada o especificación de requerimientos*.

Diseño.- En ésta fase se obtiene como diseño un modelo que representa la solución de los requerimientos del usuario. Se diseñan como serán las interfaces, programas y bases de datos, con el objetivo de cubrir los requerimientos del usuario.

Implantación.- Las especificaciones diseñadas en el paso anterior se pasan a un código. Se programan en el lenguaje apropiado, en forma estructurada y siguiendo buenos criterios de programación. Evitando programación redundante y respetando el ocultamiento de información.

Pruebas.- Se generan casos de prueba de acuerdo a diversas técnicas para probar el sistema y a los subsistemas. Se prueba cada fase del desarrollo del proyecto, se encuentran fallas y errores y se depuran o corrigen.

Instalación.- La instalación implica la ejecución de una serie de actividades que permitirán al usuario interactuar con el sistema en forma autónoma e independiente.

Mantenimiento.- Luego de instalado el sistema toda actividad que implique modificar el sistema se considera de mantenimiento. Algunas veces ocurre que algunos procedimientos y/o políticas del usuario cambian, otras veces como el sistema no ha sido correctamente analizado, diseñado y probado surgen necesidades de modificación. Las primeras se consideran normales, las segundas no deberían ocurrir si se trabaja siguiendo el método.

**Análisis Estructurado.**

El análisis estructurado proporciona técnicas y herramientas que facilitan el reconocimiento de los requerimientos e identificación de problemas. Se utilizan principalmente herramientas gráficas. Y lo que se pretende es realizar un modelo que represente al sistema actual, desde los tres puntos de vista de cualquier tipo de sistema de información (procesos-control-datos). En el Análisis Estructurado procedemos ordenadamente realizando una serie de actividades hasta llegar al diseño, ver la siguiente tabla 2.1.5.2

Etapa Actividad Técnica – Herramienta Resultado

Etapa	Actividad	Técnica – Herramienta	Resultado
Análisis Estructurado	Levantamiento	Entrevistas	Especificación Estructurada
		Cuestionarios	
		Organigramas	
		Tablas / Árboles de decisión	
		Inspección de documentos y archivos	
		Muestreos	
	Modelo Ambiental	Declaración de objetivos	
		Lista de eventos	
		Diagrama de contexto	
		Diccionario del sistema	
	Modelo de Comportamiento	Diagrama de flujo de datos	
		Diagrama Entidad - Relación	
	Balanceo	Diagrama de transición de estados	

Tabla 2.1.5.2



Levantamiento.- Es el conjunto de actividades y tareas que tiene por objetivo la obtención y recopilación de información del sistema de información existente. Para ello el analista recurre a distintas fuentes y aplica las técnicas que considera adecuadas, asegurándose que la información y datos obtenidos son correctos.

Modelo ambiental.- Este modelo tiene por objeto determinar en forma clara y precisa que es parte del sistema y que no. En éste modelo se definen las interfaces (relaciones) entre el sistema y su medio ambiente, entradas y salidas de información. Se busca modelar el exterior del sistema, para ello deberá trazarse las fronteras o límites del sistema. Se necesita saber que información entra al sistema desde el medio ambiente y que información del sistema sale al medio ambiente.

Herramientas usadas para definir el modelo ambiental:

- **Declaración de objetivos.-** Es una declaración textual, breve y concisa del propósito del sistema. Cuenta con pocas frases y no debe llevar más de un párrafo. No tiene la finalidad de describir en forma completa y detallada al sistema, lo que hace y como lo hace, sino que se espera del sistema.
- **Lista de Eventos.-** Es una lista narrativa de los estímulos que ocurren en el mundo exterior a los cuales el sistema debe responder.
- **Diagrama de contexto.-** Se compone de un único círculo (burbuja) que representa al sistema y de flechas que llegan y salen del sistema (flujos de entrada y salida), llamadas interfaces y de rectángulos que simbolizan a las entidades externas (otros sistemas del medio ambiente con los que interactúa)
- **Diccionario del Sistema (DS).-** Es un depósito de información de los elementos del DFD, es decir, de la catalogación de los flujos, procesos y almacenamientos. Los objetivos del D.S. son: Documentar las características del sistema, Manejar detalles en grandes sistemas, Dar un



significado común para todos los elementos del sistema, Localizar errores y omisiones y Validar el D.F.D. y confirmar que esté completo y correcto.

Modelo de comportamiento.- El modelo ambiental modela el exterior del sistema. El modelo de comportamiento modela el interior del sistema. Estos modelos no son cosas aisladas, todo lo contrario. El modelo de comportamiento describe como y que hace el sistema actual y las herramientas que utiliza para ello son:

- **Diagrama Flujo de Datos (DFD).**- Permite visualizar el sistema como una red de procesos funcionales, conectados entre sí por flujos de datos y almacenamientos. El DFD es un modelo lógico del sistema que representa transformaciones, procesos de datos y algo de control. Es un modelo lógico porque no depende del hardware, software, estructuras de datos, no tiene implicaciones físicas.
- **Diagrama Entidad Relación (DER).**- Es la representación (gráfica) de los datos del sistema y sus relaciones, independientemente de las funciones y procesos que operan sobre ellos. Es un modelo conceptual y semántico. Su propósito es representar que significan los datos en la realidad. Es decir, describe las características de los datos, en forma abstracta. Componentes:
 - Entidades.- Se representan mediante el rectángulo. Son cosas u objetos del mundo real, abstracto o concreto, tangible o intangible. Estos objetos, por su cantidad y características no pueden ser recordadas por el sistema, es decir en un sistema encontraremos un conjunto de entidades del mismo tipo.
 - Relaciones.- Asociación o vinculación entre dos o más entidades. En un sistema existe un conjunto de relaciones del mismo tipo. Muchas veces la relación tiene atributos propios, es decir, determinadas propiedades que solo ocurren cuando dos entidades se relacionan.



Diagrama Transición de Estados (DTE).- Herramienta que modela el comportamiento dependiente del tiempo del sistema (modela el control).
Componentes:

- Estados del sistema.- Se representan con un rectángulo. Un estado es una situación en el que se encuentra el sistema y permanece un determinado tiempo, donde el tiempo no es infinito ni cero. En la mayoría de los casos el sistema esta esperando que algo ocurra.
- Cambios de Estado.- Son aquellas reglas ordenadas, claras, precisas de un estado a otro, deben ser observables. Se representan con flechas entre los estados.

II.2 Conceptos de Base de Datos y Modelo Entidad-Relación.

Antes de existir las Bases de Datos existieron los *Sistemas de procesamiento de archivos*, basados en registros guardados en archivos, creándose respectivamente programas para su manipulación (obtener y guardar información). Estos tenían varias desventajas:

- Redundancia de información (repetición de información)
- Inconsistencia en la información.
- Falta de manejo de concurrencia (múltiples usuarios con actualizaciones al mismo tiempo).
- Dependencia entre programa con archivo.

Debido a estas desventajas en los Sistemas de procesamiento de archivos surge la necesidad de desarrollar las Bases de Datos consistentes.



Una **base de datos** o **banco de datos (BD)** es un conjunto de datos que pertenecen al mismo contexto almacenados sistemáticamente para su posterior uso, o bien, es un conjunto de información organizada (archivo lógico (uno o varios archivos físicos)).

La finalidad de una Base de datos es servir a una aplicación o más, de la mejor manera posible.

Los niveles en que se puede observar una base de datos se le conoce como Abstracción de la Información, y estos niveles son:

- *Nivel Físico:* Describe como se almacenan los datos. El encargado de éste nivel es el DBA. Administrador de la Base de Datos
- *Nivel Conceptual:* Esquematización de los datos y sus relaciones (diagramas). Los encargados de este nivel son el Analista y Desarrollador.
- *Nivel de Visión:* Abstracción mas alta, Cómo interpretan la información los usuarios finales.

Por lo que podemos definir a un **sistema de base de datos** como un conjunto de:

- Datos: Valores registrados físicamente en la BD. La información es el significado de los datos
- Hardware: Almacenamiento físico (Storage)
- Software: Programas para manejar la BD (DBMS= Sistema Manejador de Base de Datos)
- Usuarios:
 - Analista: persona encargada de esquematizar los datos y sus relaciones, hace el Diagrama Entidad Relación DER
 - Programador o Desarrollador: encargada de programar la aplicación con la BD.



- DBA: encargado del control de la BD y su administración.
- Usuarios finales: manipulan e interpretan la información.

El objetivo fundamental de un sistema de base de datos es *crear un ambiente en que sea posible guardar y recuperar información de la Base de Datos en forma eficiente.*

Las ventajas de un Sistema de Base de Datos son:

- Reducir la redundancia
- Evitar la inconsistencia
- Mantener la integridad de los datos.
- Controlar la concurrencia
- Aplicar restricciones de seguridad.
- Los datos son independientes de los programas.

Una vez concluida ésta etapa se procede con la definición del modelo de datos, que este a su vez es el resultado del análisis de la información y consiste en la representación conceptual de los datos de ésta, Debe ser claro.

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



Modelo Entidad-Relación

Éste es el modelo más utilizado en la actualidad para modelar problemas reales y administrar datos dinámicamente, que consiste en un conjunto de objetos llamados entidades y las relaciones entre ellas.

Su idea fundamental es el uso de "relaciones" y representa la estructura lógica de la Base de datos gráficamente.

Entidad: es un objeto concreto (como una persona) o abstracto (como un día festivo), es decir, es una persona, cosa o evento de interés. Una entidad está formada por un conjunto de atributos (características propias).

Relación: describe cierta interdependencia (de cualquier tipo) entre entidades. Asociación entre entidades. Existen diferentes *tipos de Cardinalidad en las relaciones*, como lo son:

- *una a una:* una instancia de la entidad A se relaciona con una y solamente una instancia de la entidad B.
- *una a muchas:* cada instancia de la entidad A se relaciona con varias instancias de la entidad B
- *muchas a una :* varias instancias de la Entidad A se relacionan con una y solamente una instancia de la entidad B
- *muchas a muchas :* cualquier instancia de la entidad A se relaciona con cualquier instancia de la entidad B

En éste modelo, el lugar y la forma en que se almacenen los datos no tienen relevancia (a diferencia de otros modelos como el jerárquico y el de red). Esto tiene la considerable ventaja de que es más fácil de entender y de utilizar para un usuario esporádico de la base de datos.



La información puede ser recuperada o almacenada mediante "consultas" que ofrecen una amplia flexibilidad y poder para administrar la información.

El lenguaje más habitual para construir las consultas a bases de datos relacionales es SQL, *Structured Query Language* o *Lenguaje Estructurado de Consultas*, durante su diseño, una base de datos relacional pasa por un proceso al que se le conoce como normalización de una base de datos, el cual es entendido como el proceso necesario para que una base de datos sea utilizada de manera óptima.

Normalización de una base de datos.

El proceso de **normalización de bases de datos** consiste en aplicar una serie de reglas a las relaciones obtenidas tras el paso del modelo entidad-relación al modelo relacional. Se basa en describir la información usando tablas. Estas tablas se intentan estructurar de forma que cumplan unos formatos llamados Formas Normales. Cuanto más alta la forma normal, más estrictos son los criterios que cumple la tabla y más fácil resulta tratarla.

Formas Normales.

- *Primera Forma Normal (1NF)*: Todos los atributos deben de tener un sólo valor, es decir, valor único para cada ocurrencia.
- *Segunda Forma Normal (2NF)*: Todo campo que no sea clave debe depender por completo de toda la clave.
- *Tercera Forma Normal (3NF)*: No hay dependencias transitivas. Un campo debe depender de la clave y no de otro campo.
- *Forma Normal de Boyce-Codd (BCNF)*: Todos los determinantes de la tabla son clave candidata.



- *Cuarta Forma Normal (4NF)*: Una fila no debe contener dos o más campos multi-valorados (aquellos que pueden contener más de un valor simultáneamente) sobre una entidad.
- *Quinta Forma Normal (5NF)*: Una tabla puede almacenar atributos dependientes a la clave sólo por unión.

Entre las ventajas de este modelo están:

- Garantiza herramientas para evitar la duplicidad de registros, a través de campos claves o llaves.
- Garantiza la integridad referencial: Así al eliminar un registro elimina todos los registros relacionados dependientes.
- Evita la redundancia de los datos.
- Evita problemas de actualización de los datos en las tablas.
- Proteger la integridad de los datos.
- Favorece la normalización por ser más comprensible y aplicable

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

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

Para ello se definen las claves o llaves:

Una **clave primaria o llave primaria** es aquella columna (pueden ser también dos columnas o más) que identifica únicamente a esa fila. La clave primaria es un identificador que va a ser único para cada fila. Se acostumbra poner la clave



primaria como la primera columna de la tabla pero esto no tiene que ser necesario, si no es más una conveniencia. Muchas veces la clave primaria es auto numérica.

Una **clave foránea** es aquella columna que existiendo como dependiente en una tabla, es a su vez clave primaria en otra tabla.

Una **clave alternativa** es aquella clave candidata que no han sido seleccionadas como clave primaria, pero que también pueden identificar de forma única a una fila dentro de una tabla.

Una **clave compuesta** es una clave que está compuesta por más de una columna.

II.3 Características, ventajas y desventajas de SQL Server 2000.

SQL Server 2000 es el sistema de administración de bases de datos relacionales de Microsoft (RDBMS), que se encarga de mantener las relaciones entre la información y la base de datos, se asegura de que la información sea almacenada correctamente, es decir, que las reglas que definen las relaciones entre los datos no sean violadas, y recupera toda la información en un punto conocido en caso de que el sistema falle. Cuenta con soporte para el lenguaje XML, y también integra soporte nativo para el lenguaje ASP. Figura 2.3.1.



Figura 2.3.1 Instalación de SQL Server 2000

SQL Server usa la arquitectura Cliente / Servidor para separar la carga de trabajo en tareas diseñadas para servidores y tareas diseñadas para computadoras cliente. El cliente es responsable de la parte lógica y de presentar la información al usuario. Generalmente, el cliente corre en una o más computadoras, aunque también puede correr en una computadora Servidor con SQL Server.

SQL Server admite los siguientes clientes:

- Windows NT Workstation.
- Windows 2000 Professional.
- Windows 98.
- Windows 95.
- Apple Macintosh.
- OS/2
- UNIX



Los recursos disponibles del servidor y las bases de datos son distribuidos y administrados por SQL Server (tales como memoria, operaciones de disco, etc.) entre las múltiples peticiones.

Las características de SQL Server 2000 incluyen:

- Escalabilidad y disponibilidad.
- Características de base de datos corporativas.
- Facilidad de instalación, distribución y utilización.
- Almacenamiento de datos.

Escalabilidad y disponibilidad.

Se puede utilizar en un intervalo de plataformas desde equipos portátiles con Windows 98, hasta grandes servidores con varios procesadores que tengan instalado Windows 2003.

Características de base de datos corporativas.

El motor de base de datos relacional de SQL Server admite las características necesarias para satisfacer los exigentes entornos de procesamiento de datos. El motor de base de datos protege la integridad de los datos a la vez que minimiza la carga de trabajo que supone la administración de miles de usuarios modificando la base de datos simultáneamente.



Facilidad de instalación, distribución y utilización.

SQL Server 2000 incluye un conjunto de herramientas administrativas y de desarrollo que mejora el proceso de instalación, distribución, administración y su uso en varios sitios, lo que permite que el uso de las bases de datos y de los almacenes de datos resulte una parte fluida de la creación de sistemas sólidos y escalables. Estas características permiten entregar con rapidez aplicaciones que los clientes pueden implementar sencillamente.

Almacenamiento de datos.

Incluye herramientas para extraer y analizar datos de resumen para el procesamiento analítico en línea. SQL Server incluye también herramientas para diseñar gráficamente las bases de datos y analizar los datos mediante preguntas en inglés. Ver figura 2.3.2.

- Soporta XML
- Soporte nativo ASP
- Tamaño máximo de la base de datos
1.048.516 Tera bytes
- Vistas indexadas
- Vistas particionadas distribuidas
- DTS
- OLAP
- English Query
- Servicios de metadata
- Data mining

Figura 2.3.2 Características de SQL Server 2000

Microsoft SQL Server 2000 puede proporcionar los servicios de bases de datos necesarios para sistemas extremadamente grandes, por ejemplo: los servidores de gran tamaño pueden tener miles de usuarios conectados a una instancia de SQL Server 2000 al mismo tiempo, dispone también de protección total para estos entornos, cuenta con medidas de seguridad que evitan problemas al tener



varios usuarios intentando actualizar los mismos datos al mismo tiempo, así mismo asigna también de manera muy eficaz los recursos disponibles, como memoria, ancho de banda de la red y E/S del disco, entre los distintos usuarios.

Los sitios de Internet extremadamente grandes pueden dividir sus datos entre varios servidores, extendiendo la carga de procesamiento entre varios equipos y permitiendo que el sitio sirva a miles de usuarios simultáneos.

Las aplicaciones diseñadas para SQL Server 2000 se pueden ejecutar en el mismo equipo donde se encuentra instalado, la aplicación se conecta a SQL Server utilizando los componentes de comunicaciones entre procesos (IPC) de Windows, tales como la memoria compartida, evitando el uso de la red, lo que permite implementar SQL Server en sistemas pequeños en los que una aplicación debe almacenar los datos localmente.

Los sitios Web de mayor tamaño y los sistemas de procesamiento de datos a nivel corporativo generan a menudo un mayor procesamiento de base de datos del que puede admitir un único equipo. En estos grandes sistemas, los servicios de base de datos vienen proporcionados por un grupo de servidores que forman una batería bien sincronizada para manejar altos volúmenes de información.

Las innovaciones permiten a SQL Server 2000 liderar algunas de las categorías de aplicaciones de más rápido crecimiento dentro del sector de las bases de datos. Entre éstas categorías se pueden mencionar el comercio electrónico, dispositivos móviles, automatización de sucursales, aplicaciones de líneas de negocio y depósitos de datos.



Ventajas de SQL Server 2000

Algunas de sus principales ventajas son:

- Facilidad de uso. Maneja un ambiente gráfico que facilita las tareas del administrador y del cliente.
- Manejo de múltiples plataformas de hardware.
- Manejo de múltiples aplicaciones de software.
- Familiar al usuario. Al estar basado en la plataforma Windows muchas de sus características visuales le son familiares al usuario.
- Confiable.
- Tolerante a fallas.
- Expone los datos como servicios en la Web.
- Integración con Internet. El motor de base de datos de SQL Server 2000 incluye compatibilidad integrada con XML,
- Escalabilidad y disponibilidad

Desventajas de SQL Server 2000

- Costo de la licencia,
- El software y el alojamiento de aplicaciones desarrolladas en SQL Server son de un costo regular en relación a otros DBMS.
- La mayoría de las empresas de hosting no soportan SQL-Server.
- La licencia se vende por procesador o por máquina.
- Únicamente opera bajo Windows.
- Para varias de las tareas administrativas se requiere de la reinicialización del servidor.



II. 4 Características, ventajas y desventajas de Visual Basic 6.

Visual Basic es un lenguaje de programación de fácil aprendizaje guiado por eventos, y centrado en un motor de formularios que facilita el rápido desarrollo de aplicaciones gráficas, posee varias bibliotecas para manejo de bases de datos, pudiendo conectar con cualquier base de datos a través de **ODBC (conectividad abierta de bases de datos)** (Informix, DBase, Access, MySQL, SQL Server, PostgreSQL ,etc) a través de **ADO (Objetos de datos Activo)**. Visual Basic constituye un **IDE (entorno de desarrollo integrado)** que ha sido empaquetado como un programa de aplicación, es decir, consiste en un editor de código (programa donde se escribe el código fuente), un depurador (programa que corrige errores en el código fuente para que pueda ser bien procesado), un intérprete (programa que traduce el código fuente línea a línea cada vez que es ejecutado el programa), y un **GUI (constructor de interfaz gráfica)** que consiste en una forma de programar en la que no es necesario escribir el código para la parte gráfica del programa, sino que puede hacerlo de forma visual.

Diseñador de entorno de datos: Es posible generar, de manera automática, conectividad entre controles y datos mediante la acción de arrastrar y colocar sobre formularios o informes.

Asistente para formularios: Sirve para generar de manera automática formularios que administran registros de tablas o consultas pertenecientes a una base de datos, hoja de cálculo u objeto

Asistente para barras de herramientas es factible incluir barras de herramientas personalizadas, donde el usuario selecciona los botones que desea visualizar durante la ejecución.



En las aplicaciones HTML: Se combinan instrucciones de Visual Basic con código HTML para controlar los eventos que se realizan con frecuencia en una página Web.

La Ventana de Vista de datos proporciona acceso a la estructura de una base de datos. Desde esta se tiene acceso al Diseñador de Consultas y diseñador de Base de datos para administrar y registrar los mismos.

Características técnicas

- PC con un procesador Pentium
- Memoria RAM recomendada de 32Mb.
- Un mínimo de disco duro de 60Mb en la instalación básica y un máximo de 300Mb en la instalación con toda la documentación on-line.
- Pantalla VGA o superior
- Unidad de CD-ROM
- Sistema Operativo Windows a partir de W95/NT

Visual Basic es el lenguaje de programación más popular desde hace ya varios años. Esta popularidad es el resultado de múltiples causas como es su facilidad de uso ya que se pueden realizar sencillos programas con las características del entorno Windows, además de poder implementar complejas aplicaciones con bases de datos, mediante el ADO.

Insertar controles del tipo ActiveX, archivos de sonido, etc, disponiendo para ello de un completo manual de ayuda.



Integración con el entorno Windows: la cual se logra por ser Visual Basic un producto Microsoft, que funcionará sobre el sistema operativo de la misma compañía. Si sobre Windows programamos con Visual Basic y accedemos a datos en SQL Server o Access, utilizando acceso a datos de Microsoft, nuestras aplicaciones al haber sido creadas por el mismo fabricante funcionan bastante bien entre sí. Y ya no hablemos de interaccionar con herramientas de Office como el Word o el Excel.

En lo que corresponde al entorno de programación, está muy optimizado y ayuda a encontrar fallos, simplifica la búsqueda de texto en el código, permite comentarios, y sobre todo herramientas de depuración que no hay en otros lenguajes.

Es un lenguaje muy potente, que a pesar de su simplicidad permite construir aplicaciones en un tiempo relativamente corto. Y el hecho de poder trabajar con formularios, módulos, componentes **OCX (control ActiveX)**, los cuales hacen referencia a instrucciones preprogramadas capaces de ser insertadas en cualquier aplicación que está siendo desarrollada, y permite al programador ahorrar tiempo debido a que en lugar de reprogramar un código anteriormente hecho, ahora únicamente se inserta; y librerías externas simplifica enormemente el trabajo al poder volver a utilizar partes de nuestro programa en nuevas aplicaciones sin tener que rehacerlo.

Visual Basic trabaja con objetos (imágenes, botones, etc.) de forma sencilla e incluso nos permite crear nuestros propios controles OCX (una caja de texto o un botón, podemos por ejemplo, personalizar nuestras cajas de texto).

Visual Basic posee un sistema de ayuda muy completo, a modo de manual con ejemplos prácticos y código fuente, aparte, claro está, de la cantidad de manuales publicados que pueden conseguirse no sólo en cualquier librería sino también en



Internet (al ser un lenguaje muy utilizado, hallaremos sin dificultad no solo documentación, sino pequeños programitas, controles, trucos, y otras cosas dignas de estudio)

Ventajas

- La facilidad del lenguaje permite crear aplicaciones para Windows en muy poco tiempo. En otras palabras, permite un desarrollo eficaz y menor inversión en tiempo que con otros lenguajes.
- Permite generar **DLL (librerías dinámicas)** ActiveX de forma nativa y Win32 (no ActiveX, sin interfaz COM) mediante una reconfiguración de su enlazador en el proceso de compilación.
- Permite la utilización de formularios tanto a partir de recursos propios así como utilizando un IDE para diseñarlos.
- Posibilidad de desarrollar y ejecutar aplicaciones de Visual Basic 6.0 en Windows Vista sin realizar cambios en la mayoría de los casos
- Se necesitan pocos recursos de hardware para poder acceder a Visual Basic 6.0
- Permite la herencia, interfaces y sobrecarga, que lo convierten en un eficaz lenguaje de programación orientado a objetos.
- Se puede crear auténticas aplicaciones multiproceso.
- Control estructurado de excepciones.
- Atributos personalizados y compatibilidad con la especificación de lenguajes comunes **CLS (Common Language Specification)**.
- Permite el control estructurado de excepciones que facilita la creación y el mantenimiento de los programas con controladores de errores robustos y completos.
- Se agrega la sobrecarga que es la capacidad de definir propiedades, métodos, procedimientos u operadores que tienen el mismo nombre pero utilizan diferentes tipos de datos.



- Se pueden utilizar atributos para proporcionar información adicional sobre elementos de programa.
- Puede utilizar formularios de Windows Forms para crear una aplicación completa del cliente mediante Visual Basic. Ésta aplicación puede obtener acceso a una amplia gama de orígenes de datos y proporcionar utilidades de visualización y edición de datos utilizando los controles de los formularios Windows Forms.

Desventajas

- En Visual Basic 6.0 y anteriores sólo existe un compilador e IDE, llamado igual que el lenguaje.
- No existe forma alguna de exportar el código a otras plataformas fuera de Windows.
- Los ejecutables generados son relativamente lentos en Visual Basic 6.0 y anteriores al ser código pseudo-interpretado.
- Por defecto permite la programación sin declaración de variables. (que puede ser sencillamente corregida escribiendo la frase Option Explicit en el encabezado de cada formulario, en cuyo caso será menester declarar todas las variables a utilizar, lo que a la postre genera código más puro).
- Su sintaxis no es **case sensitive (no distingue entre minúsculas y mayúsculas)**
- No permite programación a bajo nivel ni incrustar secciones de código en **ASM (Ensamblador)**.
- Sólo permite el uso de funciones de librerías dinámicas (DLL) stdcall.
- Fuerte dependencia de librerías y componentes en las versiones 6.0 y anteriores, lo que dificultaba la distribución de los desarrollos entre maquinas.
- Algunas funciones están indocumentadas.



- La escasa implementación de POO en Visual Basic 6.0 y anteriores no permite sacar el máximo provecho de este modelo de programación.
- El manejo de errores que tiene mediante la orden *on error* no sigue los patrones estructurados.
- No permite el manejo de memoria dinámica, punteros, etc. como parte del lenguaje.
- No avisa de ciertos errores o advertencias (se puede configurar el compilador para generar ejecutables sin los controladores de desbordamiento de enteros o las comprobaciones de límites en matrices entre otros, dejando así más de la mano del programador la tarea de controlar dichos errores)
- El tratamiento de mensajes de Windows es básico e indirecto.
- La gran gama de controles incorporados son, sin embargo en algunos casos, muy generales, lo que lleva a tener que reprogramar nuevos controles para una necesidad concreta de la aplicación.
- Los controles personalizados no mejoran la potencia de la API de Windows, y en determinados casos acudir a ésta será el único modo de conseguir el control personalizado deseado.

II.5 Conceptos del Modelo Cliente-Servidor y Redes.

La tecnología cliente/servidor es el procesamiento cooperativo de la información por medio de un conjunto de procesadores, en el cual múltiples clientes, distribuidos geográficamente, solicitan requerimientos a uno o más servidores centrales Fig. 2.5.1. Desde el punto de vista funcional, se puede definir la computación cliente/servidor como una arquitectura distribuida que permite a los usuarios finales obtener acceso a la información en forma transparente aún en entornos multiplataforma.

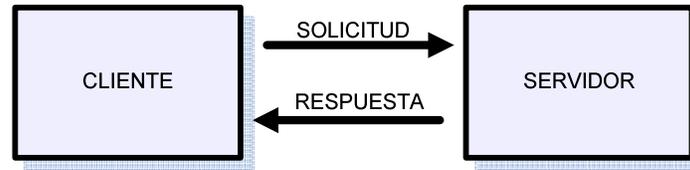


Fig. 2.5.1 Modelo Cliente-Servidor

La idea es tratar a una computadora como un instrumento, que por sí sola pueda realizar muchas tareas, pero con la consideración de que realice aquellas que son más adecuadas a sus características. Si esto se aplica tanto a clientes como servidores se entiende que la forma más estándar de aplicación y uso de sistemas clientes/servidores es mediante la explotación de las PC a través de interfaces gráficas de usuario; mientras que la administración de datos y su seguridad e integridad se deja a cargo de computadoras centrales.

Como se desprende de las definiciones anteriores, tanto clientes como servidores son entidades independientes que operan conjuntamente a través de una red para realizar una tarea.

Pero para hacer la distinción respecto de otras formas de arquitecturas o software distribuidos, se presenta una lista de características que debieran cumplir los sistemas 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 éstos los que hacen peticiones de servicios a los servidores. Éstos últimos tienen un carácter pasivo ya que esperan las peticiones de los clientes.
- No existe otra relación entre clientes y servidores que no sea la que se establece a través del intercambio de mensajes entre ambos. El mensaje es el mecanismo para la petición y entrega de solicitudes de servicio.
- Las plataformas de software y hardware entre clientes y servidores son independientes. Precisamente una de las principales ventajas de esta arquitectura es la posibilidad de conectar clientes y servidores independientemente de sus plataformas.
- 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.

Componentes del modelo Cliente/Servidor

Como se ha venido diciendo, cliente/servidor es un modelo basado en la idea del servicio, en el que el cliente es un proceso consumidor de servicios y el servidor es un proceso proveedor de servicios. Además esta relación está establecida en función del intercambio de mensajes que es el único elemento de acoplamiento entre ambos.

De estas líneas se desprenden los tres elementos fundamentales sobre los cuales se desarrollan e implantan los sistemas cliente/servidor: el proceso cliente que es quien inicia el diálogo, el proceso servidor que pasivamente espera a que lleguen peticiones de servicio y el middleware que corresponde a la interfaz que provee la conectividad entre el cliente y el servidor para poder intercambiar mensajes.



Para entender en forma más ordenada y clara los conceptos y elementos involucrados en esta tecnología se puede aplicar una descomposición o arquitectura de niveles. Esta descomposición principalmente consiste en separar los elementos estructurales de esta tecnología en función de aspectos más funcionales de la misma:

- Nivel de Presentación: Agrupa a todos los elementos asociados al componente Cliente.
- Nivel de Aplicación: Agrupa a todos los elementos asociados al componente Servidor.
- Nivel de comunicación: Agrupa a todos los elementos que hacen posible la comunicación entre los componentes Cliente y servidor.
- Nivel de base de datos: Agrupa a todas las actividades asociadas al acceso de los datos.

Cliente

El cliente es el proceso que permite al usuario formular los requerimientos y pasarlos al servidor, se lo conoce con el término front-end. Este normalmente maneja todas las funciones relacionadas con la manipulación y despliegue de datos, por lo que están desarrollados sobre plataformas que permiten construir interfaces gráficas de usuario (GUI), además de acceder a los servicios distribuidos en cualquier parte de la red.

Las funciones que lleva a cabo el proceso cliente se resumen en los siguientes puntos:

- Administrar la interfaz de usuario.
- Interactuar con el usuario.
- Procesar la lógica de la aplicación y hacer validaciones locales.



- Generar requerimientos de bases de datos.
- Recibir resultados del servidor.
- Formatear resultados.

Servidor

Es el proceso encargado de atender a múltiples clientes que hacen peticiones de algún recurso administrado por él. Al proceso servidor se le conoce con el término back-end. El servidor normalmente maneja todas las funciones relacionadas con la mayoría de las reglas del negocio y los recursos de datos.

Las funciones que lleva a cabo el proceso servidor se resumen en los siguientes puntos:

- 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

Es la interfaz que provee la conectividad entre aplicaciones clientes y aplicaciones servidoras, y entre aplicaciones y bases de datos. Es una capa de software que protege a los desarrolladores de tener que manejar detalles de bajo nivel de diferentes protocolos de comunicación, sistemas operativos y arquitecturas de bases de datos. Este tipo de interfaces incluyen API's, PRC's, Pipes, mensajería de red y accesos a bases de datos.



Construcción de aplicaciones en n capas.

A diferencia de lo que se pudiera pensar, el desarrollo en n-capas no es un producto o un estándar, es un concepto estratégico que ayuda a la construcción y despliegue lógico de un sistema distribuido.

Los sistemas de n-capas subdivididos ayudan a facilitar el desarrollo rápido de aplicaciones y su posterior despliegue, con beneficios incrementales fruto de los esfuerzos del desarrollo en paralelo coordinado y del outsourcing inteligente, resultando un enorme decremento del tiempo de desarrollo y de sus costos.

Aplicaciones de 1 capa

Las capas dentro de una arquitectura son nada más que un conjunto de servicios especializados que pueden ser accesibles por múltiples clientes y fácilmente reutilizables.

Este tipo de arquitectura se caracteriza por tener una sola asociación lógica y en ella a la presentación, la lógica de negocios y los datos; que si los ponemos como servicios se convierten en capas, ejemplos de esta arquitectura son desarrollos realizados en Excel, Access, etc.

Aplicaciones de 2 capas

Se caracterizan por tener 2 asociaciones lógicas, que prestan servicios y que al final son capas.

En la primera capa se incluye a la presentación (Interface grafica) y a la lógica de negocios, toda la lógica la escribimos en las formas (en el Click del botón por



ejemplo), y accedemos a un servicio de datos para la gestión de los mismos, por lo general a un servidor de Base de Datos.

Esta arquitectura es comúnmente llamada cliente servidor, puesto que también el programa fuente puede residir en un servidor y muchos clientes pueden acceder a él para ejecutar una copia del programa.

Aplicaciones de 3 capas

Una aplicación de tres capas es una aplicación cuya funcionalidad puede ser segmentada en tres niveles lógicos (capas):

- Los servicios de presentación.
- Los servicios de negocios (Lógica de Negocios)
- Los servicios de datos.

La capa de servicios de presentación es responsable de:

- Obtener información del usuario.
- Enviar la información del usuario a los servicios de negocios para su procesamiento.
- Recibir los resultados del procesamiento de los servicios de negocios.
- Presentar estos resultados al usuario.

El nivel de servicios de negocios es responsable de:

- Recibir la entrada del nivel de presentación.
- Interactuar con los servicios de datos para ejecutar las operaciones de negocios para lo que la aplicación fue diseñada a automatizar
- Enviar el resultado procesado al nivel de presentación.



El nivel de servicios de datos es responsable de:

- Almacenar los datos.
- Recuperar los datos.
- Mantener los datos.
- La integridad de los datos.

Al ser la primera capa un servicio, se puede inferir que las aplicaciones no solo podrían ser de escritorio, si quisiéramos que nuestra aplicación tenga una interfase Web, pues solamente bastaría con cambiar la capa de presentación y de allí en adelante nada tiene porque cambiar. Entonces nuestras páginas Web estarían alojadas en un Servidor Web las mismas que se conectan a la lógica de negocios y de allí a los servicios de datos.

Aplicaciones de 4 capas

Podríamos ir separando nuestra aplicación en mas niveles lógicos, por ejemplo, vamos a querer que nuestra aplicación tenga múltiples interfaces, es decir interfase gráfica (Standalone o Desktop) y también interfase Web.

Lo aconsejado en esta circunstancia es separar al Servidor Web encargado de alojar las páginas Web en una capa más. En este caso se tendrían 4 capas.

Mientras más servicios coloquemos a nuestra aplicación y mientras más escalable lo imaginemos, más capas lógicas van a irse añadiendo a nuestra arquitectura.

Conceptos de red

Una red es una interconexión de dos o más computadoras con el propósito de compartir información y recursos a través de un medio de comunicación, como puede ser el cable coaxial. El propósito más importante de cualquier red es enlazar entidades similares al utilizar un conjunto de protocolos que aseguren un



servicio confiable. Según el lugar y el espacio que ocupen, las redes se pueden clasificar en:

Redes LAN

Una red de área local, es la interconexión de varios ordenadores y periféricos. (LAN es la abreviatura inglesa de Local Area Network, 'red de área local'). Su extensión esta limitada físicamente a un edificio o a un entorno de pocos kilómetros.

Su aplicación más extendida es la interconexión de ordenadores personales y estaciones de trabajo en oficinas, fábricas, etc., para compartir recursos e intercambiar datos y aplicaciones.

Redes MAN

Una red de área metropolitana (Metropolitan Area Network o MAN, en inglés) es una red de alta velocidad (banda ancha), dando cobertura en un área geográfica extensa, proporciona capacidad de integración de múltiples servicios mediante la transmisión de datos, voz y vídeo, sobre medios de transmisión tales como fibra óptica y par trenzado de cobre.

El concepto de red de área metropolitana representa una evolución del concepto de red de área local a un ámbito más amplio, cubriendo áreas mayores que en algunos casos no se limitan a un entorno metropolitano sino que pueden llegar a una cobertura regional.

Redes WAN

Una Red de Área Amplia (Wide Area Network o WAN, del inglés), es un tipo de red de computadoras capaz de cubrir distancias mucho mayores a las redes MAN,



dando el servicio a un país o un continente. Su función fundamental está orientada a la interconexión de redes o equipos terminales que se encuentran ubicados a grandes distancias entre sí. Para ello cuentan con una infraestructura basada en poderosos nodos de conmutación que llevan a cabo la interconexión de dichos elementos, por lo que además fluye un gran volumen de información de manera continúa.

Topologías de redes

La arquitectura o topología de red es la disposición física en la que se conectan los nodos de una red de ordenadores o servidores, mediante la combinación de estándares y protocolos. Esta define las reglas de una red y cómo interactúan sus componentes.

Red en bus

La topología de bus tiene todos sus nodos conectados directamente a un enlace y no tiene ninguna otra conexión entre nodos. Físicamente cada host está conectado a un cable común, por lo que se pueden comunicar directamente, aunque la ruptura del cable hace que los hosts queden desconectados. Esta topología permite que todos los dispositivos de la red puedan ver todas las Señales de todos los demás dispositivos, lo cual puede ser positivo si se desea que todos los dispositivos obtengan esta información. Sin embargo, puede representar una desventaja, ya que es común que se produzcan problemas de tráfico y colisiones.

Red en anillo

El anillo es otro diseño simple basado en el bus lineal en donde los dos extremos se unen para formar un anillo cerrado, permitiendo siempre contar con 2 rutas posibles entre cada origen con cada destino dentro del anillo.



Red en estrella

Esta es la más común hoy en día y se representa a los nodos unidos punto a punto con un dispositivo central. En esta topología toda comunicación pasa siempre por un punto central llamado hub ó concentrador que repite la señal hacia el dispositivo destino.

Red en árbol

Esta topología mezcla una red tipo bus lineal como red dorsal a la que se conectan pequeñas estrellas formando una topología con forma de árbol.

Red en malla

Esta es una topología de red en la que cada nodo está conectado a uno o más de los otros nodos. De esta manera es posible llevar los mensajes de un nodo a otro por diferentes caminos.

Periféricos de Redes.

Repetidor: Estos son dispositivos conectados a dos segmentos de cable, proporcionan la amplificación y re-sincronización de las señales necesarias para conectar los segmentos.

Concentrador: Los concentradores son dispositivos similares a los repetidores, con la diferencia, que están diseñados para cableado UTP. Estos dispositivos están divididos en dos grupos:

- **Concentradores Pasivos:** Se trata de un dispositivo que centraliza el cableado de la red.



- **Concentradores Activos:** Son dispositivos que además de centralizar el cableado de la red, regeneran señales eléctricas que le llegan, realizando así, funciones de repetidor.

Bridge: Los bridge son utilizados para interconectar segmentos, un bridge se encarga de repetir paquetes. Estos son dispositivos con dos interfaces Ethernet. El puente opera sobre ambas interfaces, capturando una de las tarjetas todos los paquetes válidos y entregándolos a la siguiente, por ejemplo si el puente conecta a dos Ethernets (E1 y E2), el software toma cada paquete que llega en E1 y lo transmite a E2, y viceversa.

Los puentes son superiores a los repetidores porque estos no retransmiten errores, ruido o paquetes deformados, un paquete se reenvía cuando se tiene la seguridad que este completo.

Router: Son equipos que pueden filtrar protocolos y direcciones a la vez. Además los routers pueden interconectar redes distintas entre sí. Un router trabaja con tablas de enrutado con la información que generan los protocolos, deciden si hay que enviar un paquete o no, deciden cual es la mejor ruta para enviar un paquete.

Bridge Router: Es un dispositivo que combina las funciones de un bridge y de un ruteador. Este puede soportar múltiples protocolos de ruteo de paquetes y puede realizar bridge sobre paquetes no ruteables.

Gateway: Es un dispositivo capaz de conectar redes de distinto protocolos o protocolos incompatibles, funcionando como intermediario permitiendo la comunicación.



Capítulo III



ANÁLISIS Y PLANTEAMIENTO DEL PROBLEMA

III.1 Diagrama general del proceso e identificación de su problemática.

En este capítulo describiremos el proceso general que se lleva a cabo en la recepción-entrega de los equipos, así como los problemas que se presentan durante este proceso.

El diagrama general de proceso se representa en la figura III.1.1 en el cual el cliente realiza una solicitud y espera la solución de su problema. Para poder realizar esto, se lleva un registro del servicio, se realiza la recepción del equipo, el equipo se entrega al laboratorio y se realiza la reparación del mismo, dando así solución a la solicitud generada.

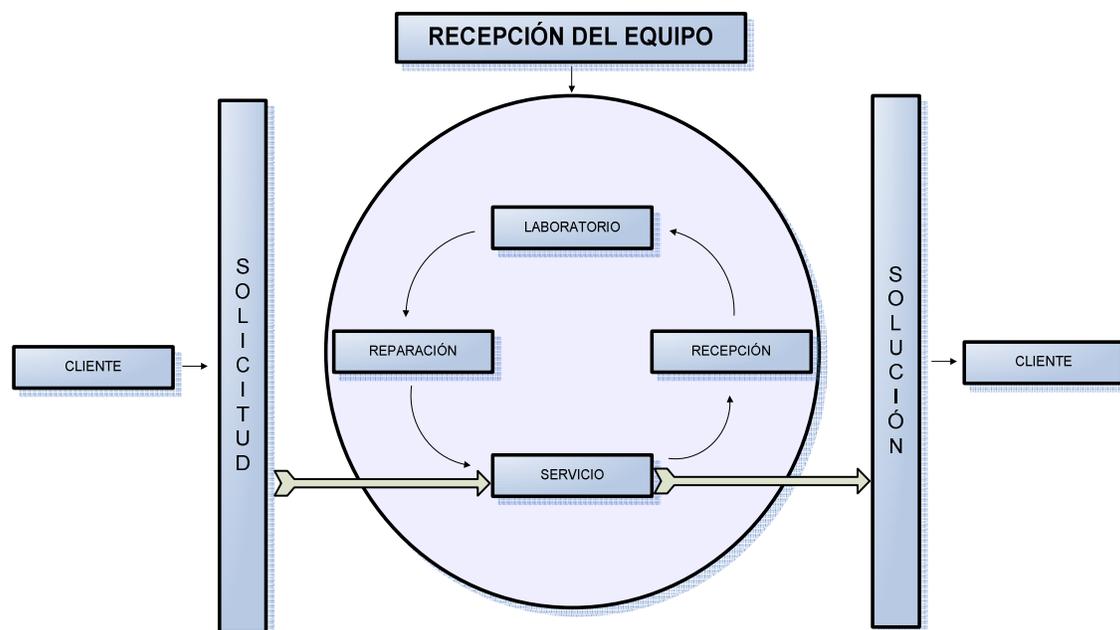


Fig. III.1.1 Diagrama General del Proceso



De este diagrama podemos deducir, como se realiza la recepción del equipo, por lo cual describiremos los procesos que se llevan a cabo para la reparación del equipo dañado.

- El proceso inicia cuando el usuario acude al centro de servicio para reportar la falla en su equipo telefónico.
- El personal de atención al cliente levanta la orden de servicio o en su defecto toma el folio de garantía, en caso que el equipo cuente con ella, así como la fecha y hora en que se realiza la solicitud. El primer problema que se presenta es que no se cuenta con una base de datos de los empleados que atienden cada solicitud, por lo que no se puede llevar un control sobre la eficiencia del personal.
- El personal de atención a clientes llena el formato de recepción del equipo, en el que se solicitan los datos del cliente, los datos del equipo y se anota la falla que presente, por último se informará al usuario del costo de la reparación o se le proporcionará una cotización. En caso de ser garantía se anota en el formato y se indica al usuario que no hay ningún costo por la reparación.

En este punto no se cuenta con un registro de clientes, por lo que si una persona ya había reportado algún equipo tiene que volver a registrar sus datos, otro problema son los errores que se presentan en el llenado de los formatos. Además, como no se cuenta con una forma de saber si un empleado dio de alta un cliente con anterioridad con el mismo equipo se podría tener información duplicada.



- Se guarda el equipo junto con los celulares dañados para su posterior traslado al laboratorio para reparación. En este caso los registros que se tienen de estos equipos están en papel, por lo que realizar una búsqueda de información sobre algún equipo es muy tardado.
- Se realiza el envío de los equipos dañados al laboratorio para su reparación.
- Los equipos son entregados en el laboratorio con la orden de servicio y/o el folio de garantía. Una vez que el equipo ingresa al laboratorio no se puede conocer el estado de reparación en que se encuentra el celular.
- Se realiza la revisión de cada equipo telefónico para solucionar la falla. Aquí el problema radica en la falta de catálogos que manejan las distintas compañías para modelos telefónicos, reparaciones y partes de los equipos. Otro problema es que no se cuenta con un indicador de cuantos celulares están en reparación.
- Solucionada la falla el técnico llena el formato de laboratorio en el que se describe las fallas que se encontraron, la solución a las mismas, la salida programada del equipo y el nombre del técnico que reviso el equipo, para que sea devuelto al centro de atención a clientes. En este caso, los registros que se tienen están en papel y el llenado puede presentar errores por parte del técnico. Además, no se lleva un histórico de cuantas veces un celular ha ingresado al laboratorio.
- Los equipos se entregan en el centro de atención a clientes, con el formato de laboratorio para su registro. El control que se lleva de las composuras es mediante papel.



- Se le informa al usuario de la reparación del equipo, notificándole la fecha y hora de su entrega.
- Se realiza el llenado del formato entrega de equipo donde se indica el nombre del cliente, los datos del equipo, la solución de la falla y el costo de la reparación o la solución debido a la garantía del equipo. El problema que se presenta es la compra de recibos de pago. También se tienen errores en el llenado del formato de entrega.

La figura III.1.2 muestra el proceso de recepción-entrega del equipo.

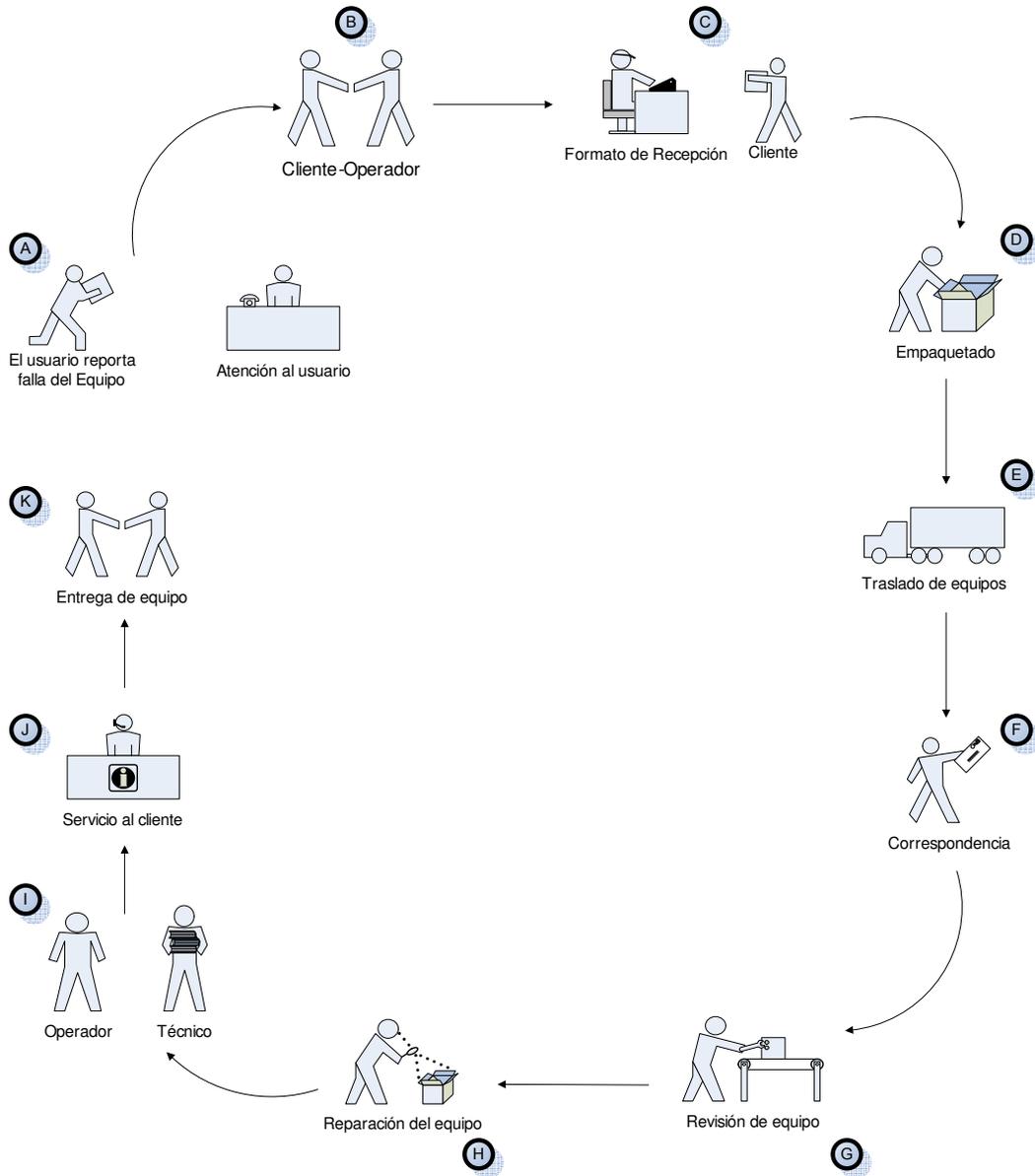


Fig. III.1.2 Recepción entrega del equipo



III.2 Integración de requerimientos generales y particulares por cada módulo.

III.2.1 Requerimientos generales del sistema.

Para el desarrollo óptimo de nuestro sistema, definiremos los requerimientos que nos permitirán cubrir todas las necesidades existentes:

- Se desea un sistema simple de manejar con la finalidad de no requerir de especialistas en su operación, si no que al contrario, cualquier persona con una mínima capacitación sea capaz de utilizarlo.
- Para facilitar su manejo debe de contar con interfaces gráficas amigables con el usuario.
- Al contar con interfaces gráficas es necesario que sea posible utilizarlo con el mouse, el teclado y el manejo de íconos.
- Se requiere que el sistema sea capaz de interactuar de alguna manera con los formatos o archivos con los que se proporciona información para la actualización de catálogos de nuevos modelos de equipos.
- Controlar el número de reportes que se soliciten.
- Obtener comprobantes de servicio para la empresa y el cliente.
- Agilizar el proceso de entrega-recepción del teléfono celular.
- Contar con un archivo histórico de fallas (bitácora) generadas en los equipos.
- Contar con una base de datos propia que permita llevar un histórico de cada equipo recibido, reparado y puesto otra vez en servicio, así como de los irreparables.
- Elaborar un diagnóstico de las marcas mas confiables en el mercado.
- Reducir la cantidad de información requerida al momento de recibir el teléfono celular ya que esta deberá estar contenida en catálogos.
- Validación del usuario, así como niveles de acceso



III.2.2 Requerimientos particulares de área.

Área Administrativa

El sistema requiere de un módulo que nos permita gestionar de una manera óptima y eficiente la información con la que se cuenta.

- Información del personal que labora en la empresa
- Información de los clientes
- Administración de los catálogos que contienen las fallas más frecuentes
- Especificar los parámetros respectivos de las órdenes de servicio vencidas.
- Alta, baja y actualización de usuarios del sistema
- Cambio de contraseña para poder acceder al sistema
- Respaldo de la base datos

Atención al cliente

El área de atención al cliente necesita de un sistema fiable para el levantamiento de ordenes de servicio, así como para la consulta del status de los equipos que ingresaron para revisión al laboratorio y si estos están en garantía o fuera de ella, pero que sobre todo este proceso sea de manera ágil al realizar la consulta y funciones como:

- Consulta de información referente al status del equipo.
- Verificación de tiempo de garantía.
- Captura mínima de datos en el levantamiento de órdenes de servicio.
- Generación de comprobante de orden de servicio.



- Reportes pendientes por atender

Laboratorio

El área de laboratorio requiere de un módulo del sistema en el que pueda ingresar un diagnóstico del equipo, una vez realizada una revisión técnica de la falla o posible falla, así como un presupuesto de la reparación si es que el equipo no entró en garantía.

- Orden de servicio
- Reporte general de fallas
- Folio de garantía
- Tiempo requerido para la reparación
- Catálogo con las fallas más frecuentes
- Nombre del técnico que realiza el servicio
- Cierre de la orden de servicio

Facturación y cobranza

Si la reparación del equipo se realizó mediante el respectivo cargo por no entrar en garantía, el área de facturación generará esta.

- Notificar el costo de la reparación del teléfono
- Mostrar en pantalla toda la información del cliente, de ser necesario actualizar la información.
- Expedición de recibos de pago o facturas para la empresa y el cliente.
- Permitir la actualización de pagos e identificar aquellas cuentas que aun no han sido saldadas



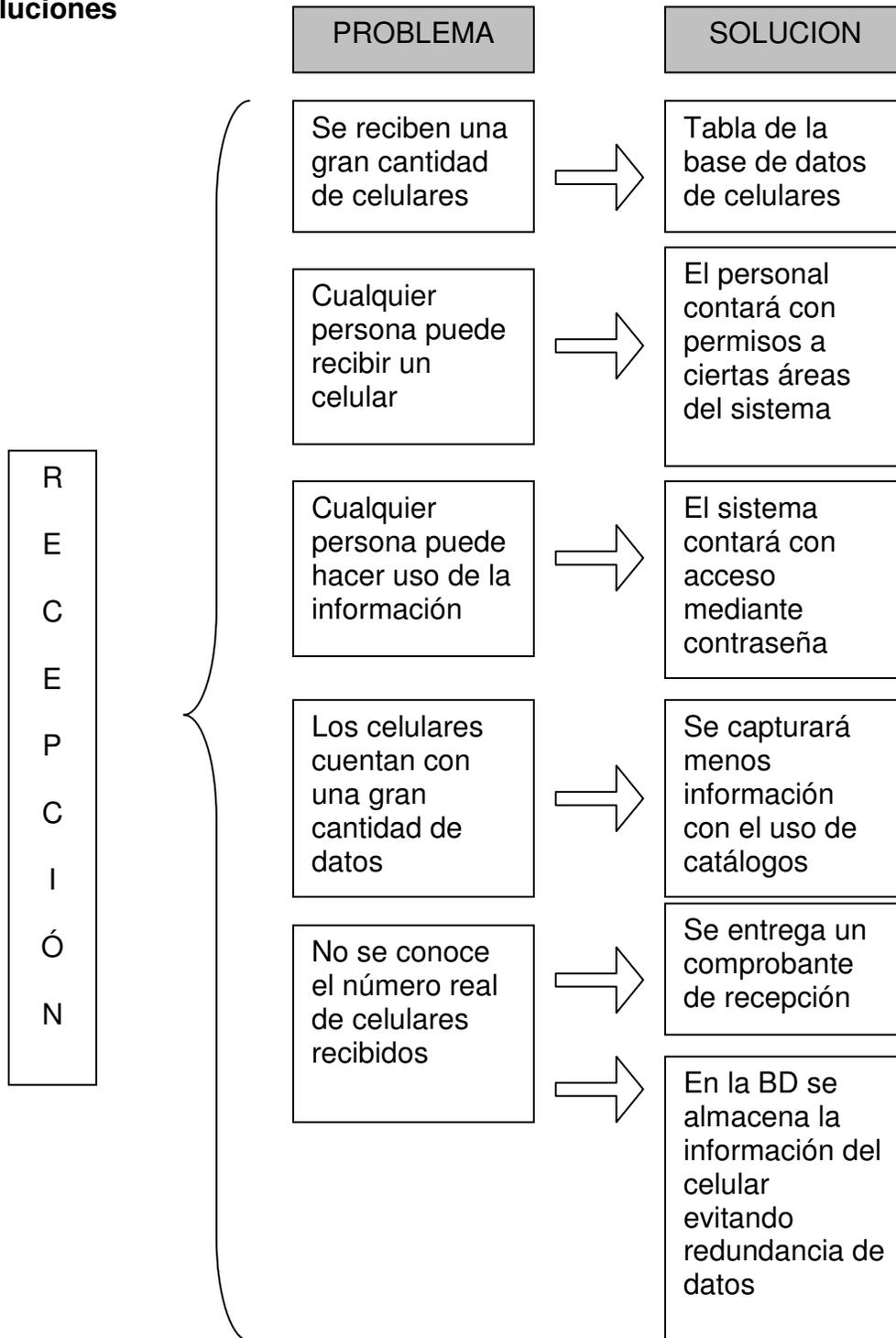
Área de Entrega

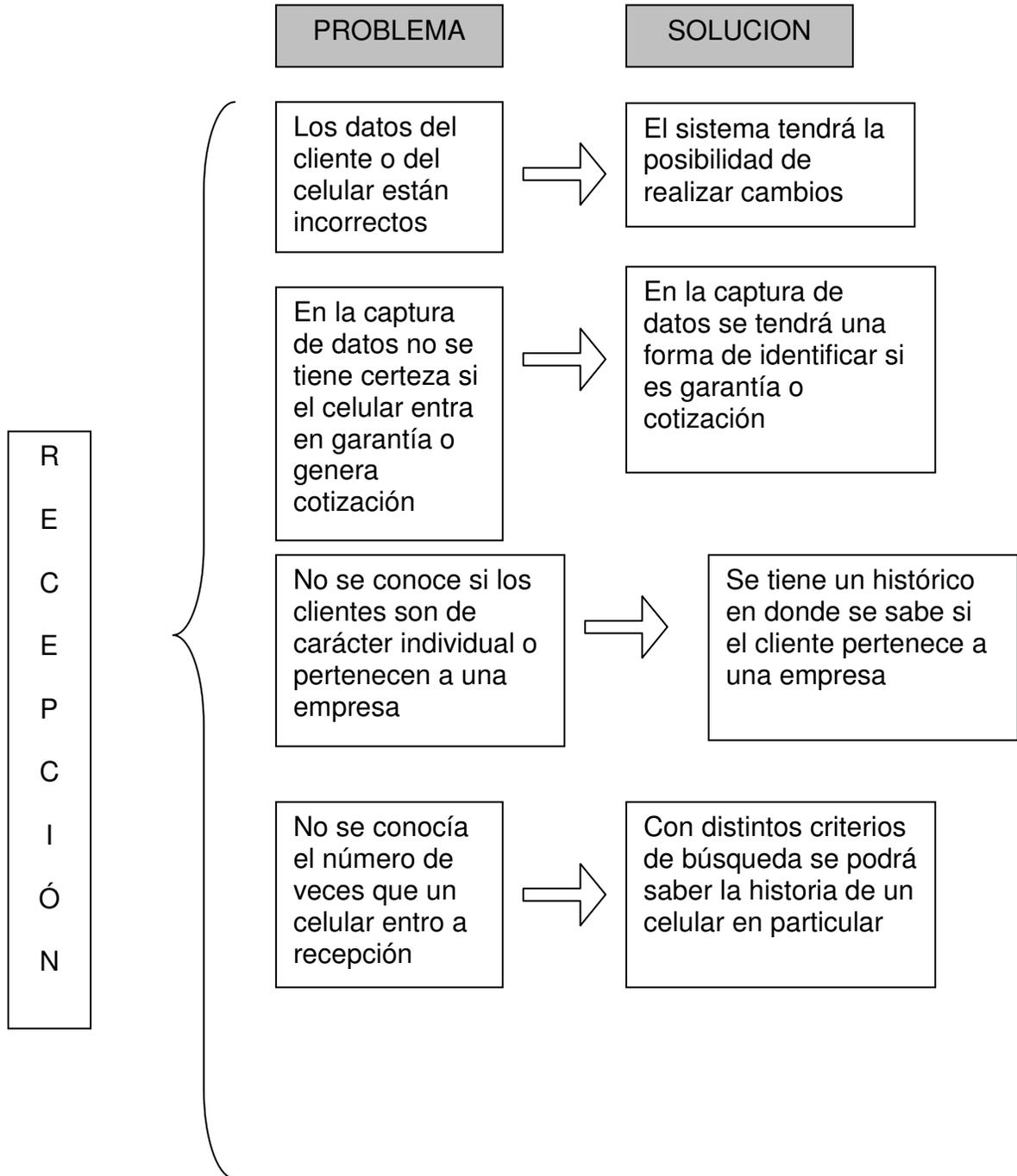
Una vez reparado el equipo se procede a la entrega del mismo al cliente, en este módulo se debe de contar con:

- Desplegar el número de la orden de servicio, el folio de la garantía, las especificaciones del tipo de teléfono, el tipo de falla y las observaciones pertinentes
- Hora de salida del equipo, número de recibo, nombre de la persona que recibe.
- Forma de pago (tarjeta de crédito o efectivo)



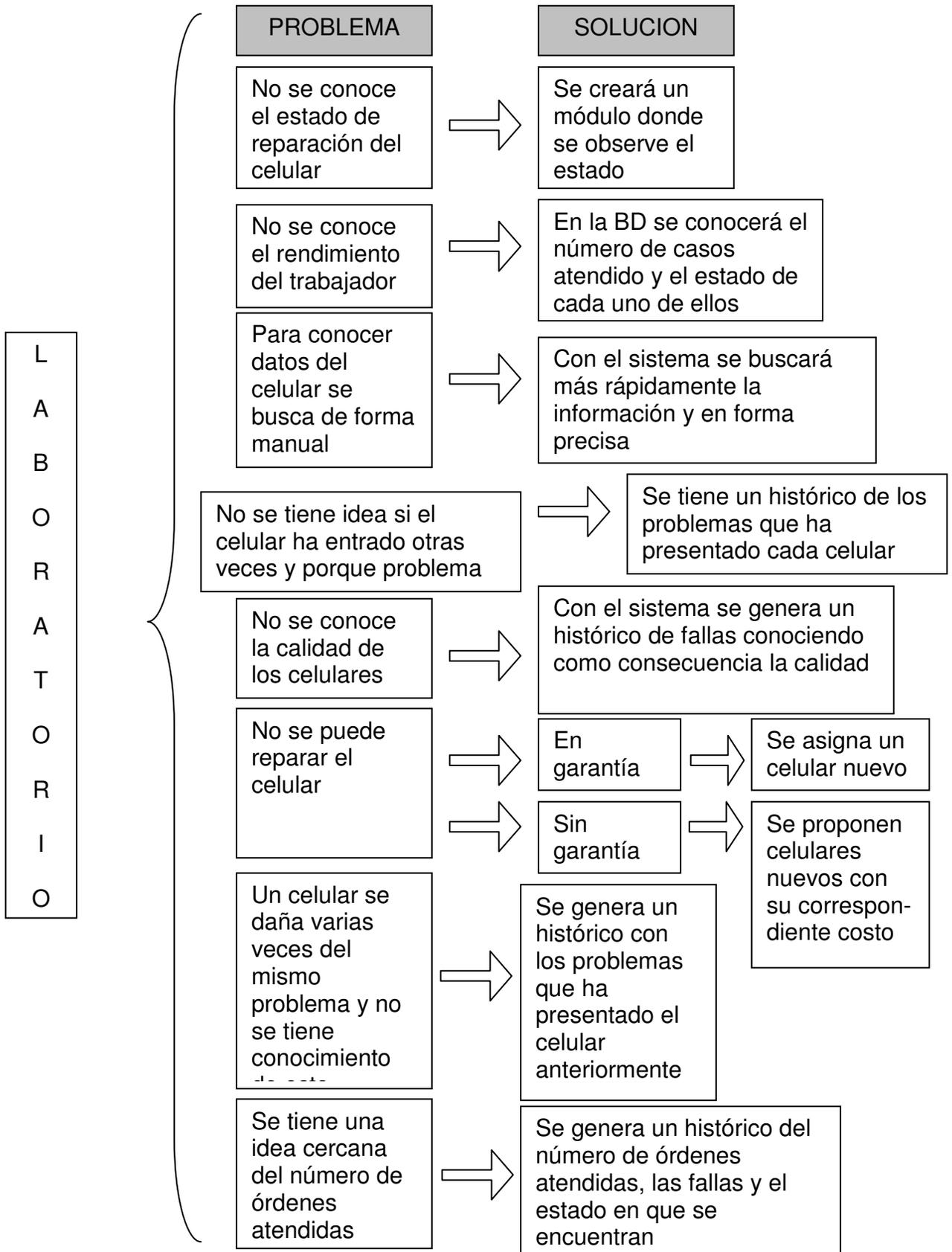
III.3 Elaboración de un diagrama de bloques y sugerencias de posibles soluciones

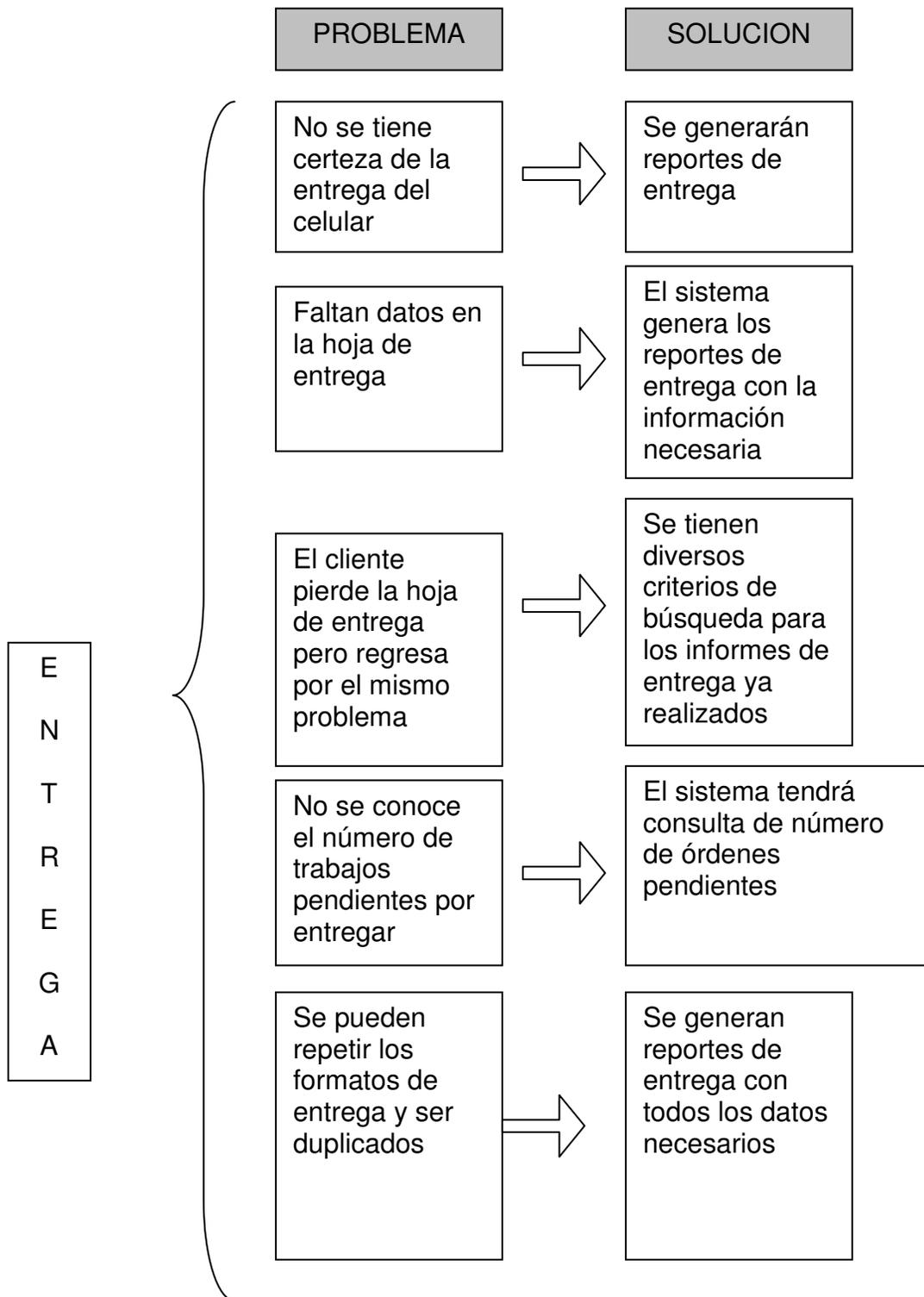






Análisis y Planteamiento del Problema







III.4 Análisis de la información recabada para la generación del posible diccionario de datos.

Los procesos del manejo y organización de la información en el momento que se reciben los teléfonos celulares en la empresa arrojan datos importantes para su canalización y en especial para el resguardo de estos mientras son o no atendidos en alguna reparación, esto permite conocer a quien le pertenece el teléfono celular una vez ingresado, la descripción del mismo, como sus características telefónicas, partes, estado en el que llega, posible(s) falla(s) y demás detalles importantes.

Para llevar el control de la información dentro de la empresa es necesario definir los procesos en los que se puede distribuir la canalización del teléfono celular y las características del mismo, estos se identifican en 3 etapas:

- Recibo de entrada
- Diagnóstico de reparación o atención en laboratorio.
- Recibo de salida.

Recibo de Entrada.

En esta etapa es importante controlar una orden de servicio, la fecha de ingreso del teléfono a la empresa, a quien le pertenece el teléfono celular, registrar datos importantes, como: Nombre del cliente o de la empresa, dirección o domicilio (calle, número, código postal, delegación o municipio, teléfono, etc.), así mismo, las características del teléfono, marca, tecnología, modelo y proveedor, como también accesorios adicionales, por ejemplo: baterías, cargadores, gomas, antenas, tapas, auriculares, tarjetas de memoria y/o SIM, manos libres, funda, etc.



Luego de que el teléfono ha sido ingresado a la empresa y dado que los registros son importantes, se procede a recabar información mas detallada en el diagnóstico de reparación o atención en laboratorio.

Diagnóstico de reparación o atención en laboratorio.

Una vez ingresado a la empresa y teniendo a grandes rasgos características del teléfono y su(s) posible(s) falla(s), entra en la etapa de diagnóstico de reparación o atención en laboratorio, la cual un técnico especializado emitirá un reporte detallado de la(s) falla(s) y la(s) posible(s) solución(es) del mismo.

El diagnóstico elaborado por el técnico, requiere de conocer los datos del cliente y la orden de servicio para llevar el control dentro del laboratorio. Este diagnóstico, pretende dar a conocer con exactitud todas y cada una de las partes físicas que pudieran ser reemplazadas del teléfono celular, por ello se identifican las partes involucradas: tapas, baterías, gomas, auriculares, cargadores, antenas, tarjetas de memoria y/o SIM, los proveedores involucrados de cada una de las partes, posibles fallas del software instalado, los modelos existentes del teléfono por proveedor, etc.

Así mismo, dar una descripción detallada de la(s) falla(s) en el teléfono celular, personal que haya realizado el diagnóstico o reparación del mismo, sin descartar el costo de la misma reparación incluyendo costos de las partes reemplazadas y la fecha de realización y termino del mismo.

Al finalizar el diagnóstico el técnico notifica a la recepción del término de la reparación, por lo que lo canaliza con una fecha de salida al personal encargado de éste, que ha finalizando un proceso más y se envía al proceso de salida.



Recibo de Salida.

Es indispensable dar a conocer al cliente a la empresa los detalles de la reparación o el diagnóstico de la falla(s) del teléfono celular una vez encontrado éste en el proceso de salida.

Para la empresa es indispensable tener conocimiento de todo lo realizado al teléfono celular en su recibo de salida, por lo que es necesario que la orden de salida deba de contener los datos relacionados al mismo, como por ejemplo: la orden de servicio, fecha de ingreso y salida, características del teléfono celular, modelo, tecnología, partes involucradas en la reparación, descripción breve de la falla, la descripción de lo que se le haya realizado para su reparación y quien lo realizó.



III.5 Ubicación de la solución a desarrollar para el back-end y el front-end

De acuerdo a los diferentes tipos de herramientas para desarrollo de sistemas y bases de datos, debemos analizar cada una de las opciones para poder utilizar la que mejor se adapte a nuestras necesidades y alcances; por un lado, analizando las características del front-End, y por el otro, el back-End.

En muchos casos, los términos front- End y back-End se refieren al principio y final de un proceso. Estos términos adquieren una relevancia mayor en ciertas áreas particulares. En nuestro caso, el diseño de software, el front-end es la parte del software que interactúa con el usuario y el back-end es la parte que procesa la entrada desde el front-end. La idea general es que el front-end es el responsable de recolectar los datos de entrada del usuario, que pueden ser de muchas y variadas formas y procesarlas de una manera conforme a la especificación que el back-end pueda usar. La conexión del front-end y el back-end es un tipo de interfaz.

En el diseño y/o desarrollo del sistema el front-end hace referencia a la visualización del usuario, y back-end al administrador de la base de datos con sus respectivos sistemas. La arquitectura cliente – servidor es una forma de dividir y especializar programas y equipos de computo a fin de que la tarea que cada uno de ellos realiza se efectúe con la mayor eficiencia y permita simplificarlas. En el modelo cliente – servidor, el trabajo se reparte entre “dos computadoras”.

3.5.1 Ventajas de la arquitectura cliente – servidor

El servidor no necesita tanto trabajo en el procesamiento parte del proceso se reparte con los clientes, además se reduce el tráfico de la red considerablemente. Idealmente 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; además este tipo de arquitectura aporta independencia entre las aplicaciones que usan el mismo servicio, de esta forma los cambios a lo largo del tiempo en uno no deben afectar a los otros; esta flexibilidad es cada vez más importante, dado que la tendencia a construir grandes aplicaciones a partir de componentes distribuidas más pequeñas es cada vez mas utilizada.

3.5.2 Interfaz Front – End.

Es una aplicación donde los usuarios interactúan directamente con las funciones del sistema, cubre todas las interfaces con las cuales un usuario interactúa con los sistemas, ya sean locales o remotos, sus funciones principales son:

- Diseño de formatos.
- Presentación.
- Lógica de la aplicación.
- Manipulación de datos.
- Herramientas de consulta.
- Utilerias / menus

Puede haber argumentos a favor de una u otra tecnología en el desarrollo de sistemas Front - End, y probablemente del mismo peso y defendidos por los diferentes desarrolladores. La discusión se puede simplificar un poco más en el uso de tecnologías propietarias o abiertas. Por eso, todos los argumentos que se puedan dar a favor de una u otra tecnología debe conducir a utilizar la herramienta que mejor pueda ayudar a resolver el problema concreto con el que alguien se encuentre; en primer lugar, conociendo las ventajas e inconvenientes de cada solución, sabiendo qué se puede y qué no se puede hacer y qué perspectiva de futuro inmediato hay. A continuación veremos las características de algunas tecnologías. Ver la tabla 3.5.2.1



Característica	Visual 6.0	C++	Delphi
Facilidades para manejo de interfaz de usuario	Excelente	Excelente	Bueno
Rapidez	Excelente	Excelente	Excelente
Componentes Active X	Excelente	Excelente	Bueno
Manejo de bases de Datos e interfaz	Excelente	Excelente	Bueno
Componentes Visuales para el desarrollo de sistemas	Bueno	Excelente	Bueno
Herramienta para el desarrollo de servicios web	Excelente	Excelente	Excelente
Elaboración de pantallas	Bueno	Bueno	Bueno
Costo	Excelente	Regular	Bueno
Conectividad a diferentes bases de datos	Excelente	Excelente	Bueno
Desempeño	Bueno	Bueno	Bueno
Soporte	Bueno	Bueno	Bueno
Rediseño	Bueno	Excelente	Regular
Documentación	Excelente	Bueno	Bueno
Depuración de control de errores	Excelente	Bueno	Regular
Facilidad de aprendizaje	Excelente	Bueno	Bueno

Tabla 3.5.2.1 Comparativo de Herramientas de desarrollo

3.5.3 Back – End

Es un conjunto de elementos (programas) que sirven como complemento de una interfaz Front-End. Ayuda en la administración, control y configuración de los sistemas teniendo un acceso directo a los recursos (base de datos, comunicaciones, servidores, etc.), que el sistema requiere, entre sus funciones principales se tienen:

- Administración de la memoria.
- Seguridad.
- Manejo de base de datos.
- Procesamiento remoto.



Los lenguajes del lado del servidor se ejecutan cuando una página es solicitada por parte de un cliente, de algunos de ellos tenemos las siguientes propiedades que permiten obtener la mejor decisión para utilizar en el desarrollo de un sistema de acuerdo a nuestras necesidades y alcances. En la tabla 3.5.3.1 presentaremos las características generales de algunos manejadores de bases de datos.

Característica	SQL Server 2000	Progress	Informix
Robustez	Excelente	Excelente	Excelente
Variedad de plataformas en las que opera (multiplataforma)	Regular	Excelente	Regular
Disponibilidad de herramientas	Bueno	Bueno	Bueno
Integración de datos	Excelente	Bueno	Bueno
Compatibilidad con otros lenguajes de desarrollo	Excelente	Bueno	Bueno
Disponibilidad de herramientas	Bueno	Bueno	Bueno
Integración de datos	Excelente	Bueno	Bueno
Capacidad de almacenamiento de información	Excelente	Bueno	Bueno
Factibilidad de accesos múltiples	Excelente	Bueno	Bueno
Restauración del sistema	Bueno	Bueno	Bueno
Cantidad de clientes que pueden atender (conurrencia)	Bueno	Regular	Regular
Tiempo de importación y generación de índices	Bueno	Bueno	Regular
Integridad de datos	Excelente	Excelente	Bueno
Facilidad de uso	Bueno	Regular	Regular
Calidad de los reportes	Excelente	Bueno	Regular
Facilidad de programación	Bueno	Bueno	Bueno
Compartir los datos con otras bases de datos	Excelente	Bueno	Regular
Diversidad de tipos de datos	Excelente	Excelente	Bueno

Tabla 3.5.3.1 Comparativo de Manejadores de Bases de Datos



3.5.4 Ubicación de la Solución

La base de datos y la herramienta visual de desarrollo que se ha decidido utilizar para la elaboración del proyecto son SQL Server 2000 (Back-End) y Visual 6.0 (Front-End) debido a sus características y a la gran relación que existe entre ellos, al ser estos productos de Microsoft.

La elección de la base de datos que se va a utilizar es un tema importante y la decisión debe ser fundamentada en cual de las diferentes opciones de las que disponemos se adapta mejor a nuestras necesidades.

Es por esto que enumeramos las razones por las cuales se eligió SQL Server 2000:

- Facilidad para el uso de la herramienta
- Rapidez de la aplicación
- Capacidad de almacenamiento de la información
- Restauración del sistema
- Integridad de datos

SQL Server 2000 cumple de manera satisfactoria con los requisitos mas importantes a considerar para lograr con esto la elección óptima del producto. SQL Server 2000 es un manejador poderoso y fácil de usar, y se cuenta con la licencia para utilizarlo; es el que posee el usuario y del cual posee más experiencia. Además no esta dispuesto a invertir en otro producto.

La herramienta para elegir del Front-End se eligió respecto a los siguientes parámetros:

- Facilidad de manejo de la interfaz



- Conectividad completa con todos los formatos de datos
- Generador de reportes para la edición rápida y precisa de éstos.
- Acceso con manejadores de bases de datos
- Conocimientos de la herramienta
- Fácil ejecución

Visual 6.0 cumple de manera satisfactoria con los requisitos mas importantes a considerar para lograr con esto la elección óptima del producto. Además, la facilidad de aprendizaje y la interfaz por sí misma que proporciona nos da la pauta para la mejor utilización y el mejor desarrollo de la programación, aunado a la obtención de información y ayuda por diversos medios; lo mismo que en el anterior es el que posee el usuario; y del cual posee más experiencia. Y no esta dispuesto a invertir en otro producto.



Capítulo IV



ANÁLISIS, DISEÑO Y CONSTRUCCIÓN DEL SISTEMA

IV.1 Justificación de metodología a utilizar.

IV.1.1 ¿Por qué utilizar la Metodología Yourdon DeMarco?

La Ingeniería del Software comprende las técnicas y procedimientos ingenieriles para el desarrollo del software.

La Ingeniería de Software no se plantea sólo como una actividad de programación, previamente son necesarias las fases de análisis y diseño y posteriormente la integración y la verificación, incluso el mantenimiento cuando el producto ya está en explotación. (**Ciclo de vida**).

Inicialmente la tarea de desarrollo era realizada individualmente por hábiles creativos, de forma poco disciplinada. El trabajo en equipo supone la división y organización del trabajo utilizando **metodologías de desarrollo**.

Se han elaborado metodologías que definen los pasos a seguir y la documentación a producir como resultado del proceso.

Las metodologías de desarrollo que se utilizan cada vez con mayor frecuencia, herramientas CASE (Computer Aided Software Engineering o Ingeniería de Software Asistida por Computadora).

Estas son aplicaciones de computadora para la automatización de etapas del desarrollo (y mantenimiento) de sistemas de información. Las herramientas CASE se agrupan en dos clases:



- Upper CASE para automatizar la primera etapa del desarrollo: Planificación y Diseño.
- Lower CASE orientadas a la generación de códigos y mantenimiento (Programación).

Cada Metodología se basa en la definición de un conjunto de bases o etapas en el ciclo de vida de un Sistema.

Las Metodologías de desarrollo más comunes son la francesa Merisse, la inglesa SSADAM y la americana basadas en teorías de Yourdon y De Marco.

Para Merisse hay cuatro fases como se puede ver en la figura IV.1.1.

Estudio Preliminar, Estudio Detallado, Realización y Puesta en Marcha.

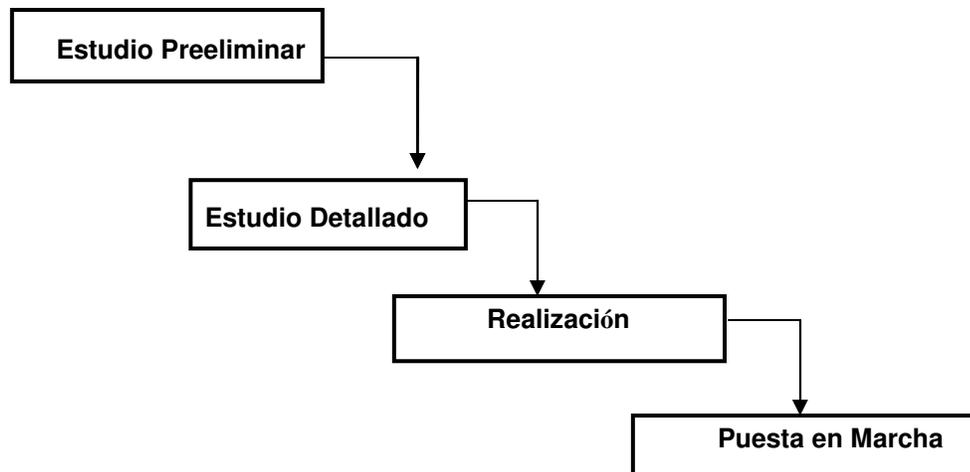


Figura IV.1.1 Metodología Merisse



En la Metodología SSADAM hay tres fases como se puede ver en la figura IV.1.2.

Estudio de Factibilidad, Análisis y Diseño.

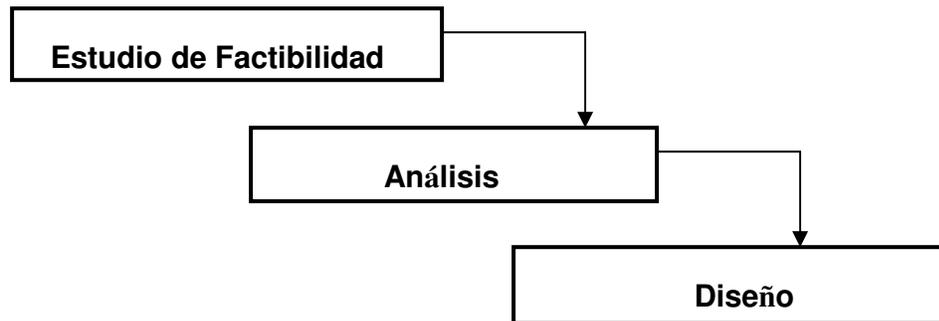


Figura IV.1.2 Metodología SSADAM

En la metodología Yourdon hay cinco fases:

Especificaciones, Análisis Lógico, Diseño Físico, Implantación y Mantenimiento, ver figura IV.1.3.

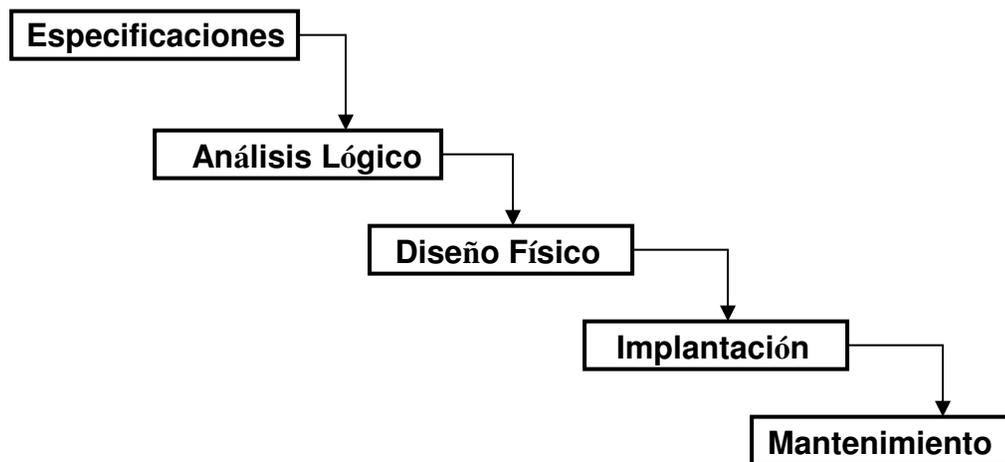


Figura IV.1.3 Metodología Yourdon DeMarco



IV.2.1 Metodología Estructurada

El análisis estructurado es el nombre con el que se designa a una clase de métodos que analizan un problema mediante la construcción de modelos de flujo de datos como es la metodología Yourdon DeMarco.

De acuerdo a la primera definición operativa hecha por DeMarco, el análisis estructurado es el uso de las siguientes técnicas para producir una especificación del sistema requerido:

- Diagramas de Flujo de Datos
- Diccionario de Datos
- Inglés Estructurado
- Tablas de Decisión
- Árboles de Decisión

Estas técnicas son adecuadas para el análisis de Sistemas de Información. El desarrollo del análisis estructurado para 'tiempo real' o 'sistemas insertos' [embedded systems], ha sido ampliado con la siguiente lista:

- Esquema de transformación
- Diagramas de Transición de Estados
- Listas de Eventos
- Esquema de Datos
- Especificaciones de Precondición- Postcondición

Los métodos del análisis estructurado caen dentro de los grupos mostrados en la **Tabla IV.2.1** tomando en cuenta si identifican, organizan o especifican a las funciones o entidades.



Actividad		Técnica
Identificación de Funciones	de	Listas de Eventos Historias de Vida de Entidades
Organización de Funciones	de	Diagramas de Flujo de Datos Esquema de Transformación Actigramas
Especificación de Funciones	de	Inglés Estructurado Tablas de Decisión Árboles de Decisión Diagramas de Transición de Estado Tablas de Transición Precondiciones-Postcondiciones
Identificación de Entidades		"Identifique los sustantivos en la descripción"
Organización de Entidades		Diagramas de Estructura de Datos Esquema de Datos Diagramas de Entidad-Relación
Especificación de Entidades	de	Diccionario de Datos

Tabla IV.2.1 Técnicas de Análisis Estructurado

El diseño estructurado es una disciplina enfocada al diseño de software que se adhiere a un conjunto específico de reglas basadas en principios tales como el diseño top-down [deductivo], el perfeccionamiento cuidadoso y el análisis del flujo de datos. En el enfoque de Yourdon De Marco, la técnica puede estar caracterizada por el uso de:

- Estructura de caracteres que muestran la jerarquía, control y flujo de datos;
- Pseudocódigo para describir el procesamiento.



La dirección bien definida entre las fases de análisis y diseño es una ventaja mayor que los métodos estructurados. La versión de DeMarco del análisis estructurado define cuatro tipos de modelos. La tabla IV.2.2 relaciona los modelos de los estándares de la ingeniería de software en la fase del ciclo de vida en la cual deben ser construidos.

Modelo No.	Nombre del Modelo	Fase de los Estándares de la Ing. de Software
1	modelo físico en curso	Fase SR
2	modelo lógico en curso	Fase SR
3	modelo lógico exigido	Fase SR
4	modelo físico exigido	Fase AD

Tabla IV.2.2 Modelo de Análisis Estructurado

Se puede ver que la metodología que utilizamos en nuestro sistema cumple con el ciclo de vida y cubre las necesidades del sistema, ya que es el proceso de desarrollo y mantenimiento del software en general. Según el modelo elegido se describen un conjunto de actividades para llevar a cabo el ciclo de vida completo.

Algunas metodologías consideran en la fase de mantenimiento actividades como la reingeniería y la ingeniería inversa.

La Reingeniería:

Consiste en la recodificación de un programa existente eliminando las funciones fuera de uso y optimizando su estructura. La generación de estos programas puede hacerse de manera automática a través de una herramienta CASE.



Ingeniería Inversa:

Consiste en obtener las especificaciones de un programa existente a partir de su código a fin de documentarlo y darle mantenimiento. También se puede utilizar herramientas CASE.

Las metodologías utilizan formularios y gráficos o herramientas CASE para registrar la Información de cada actividad definida en la metodología y producir como resultado el desarrollo (o mantenimiento) del sistema y su documentación.

La metodología que utilizaremos para el desarrollo del sistema será el Método de Yourdon DeMarco, ya que cumple todos los puntos necesarios para el desarrollo de nuestro sistema:

Investigación preliminar: el cliente nos presenta un problema y se busca resolverlo

Determinación de los requisitos del sistema: de forma conjunta con el cliente se busca llegar a un acuerdo en como se resolverá dicho problema.

Diseño del sistema (Diseño Lógico): Se darán los detalles que establecen la forma en la que el sistema cumplirá con los requerimientos identificados durante la fase de análisis.

Desarrollo de software (Diseño Físico): Ya teniendo el software a utilizar el grupo de trabajo inicia la programación.

Prueba del sistema: Básicamente se busca que el sistema sea consistente y no produzca resultados inesperados.

Implantación, evaluación y mantenimiento: Se da al cliente para que lo pruebe y haga saber si cumple con la resolución del problema.



IV.2 Análisis y diseño del back-end

Para el análisis del sistema es indispensable contar con modelos gráficos que puedan mostrar las fronteras y la información que se desea usar, lo cual permite clarificar y comprender las necesidades a cubrir por el sistema, por lo que a continuación se analizan con detalle todas las partes que conforman el sistema; para ello se elaborarán el Diagrama de contexto, Diagrama de flujo de procesos, el Diagrama de flujo de datos, el diccionario de datos, el diagrama entidad-relación, la normalización de la base de datos y la construcción física de la base de datos.

IV.2.1 Diagrama de Contexto

El diagrama de contexto nos permitirá identificar a las personas o entidades que se comunican con el sistema y recibirá datos del medio ambiente que serán sus entradas necesarias para producir datos que el mundo exterior espera e identificará como salida.

El sistema debe reconocer cada flujo de entrada para identificar que ha ocurrido un evento y cada evento debe generar salidas inmediatas como respuesta o bien almacenar datos, que posteriormente se convertirán en salida o un indicador de un cambio dentro del sistema.

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

Los elementos que se emplean para elaborar este diagrama se muestran y describen en la figura IV.2.1.1



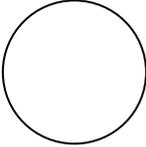
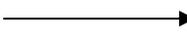
SIMBOLO	SIGNIFICADO
	Un proceso significa que se realizan algunas acciones o grupo de acciones que transformarán o modificarán la información o datos.
	Una entidad es una persona, grupo, departamento o cualquier sistema que recibe u origina información o datos.
	Un flujo de datos muestra que es transferida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

Figura IV.2.1.1 Notación gráfica usada en un diagrama de contexto

A continuación se muestra en la figura IV.2.1.2. El diagrama de contexto de nuestro sistema:

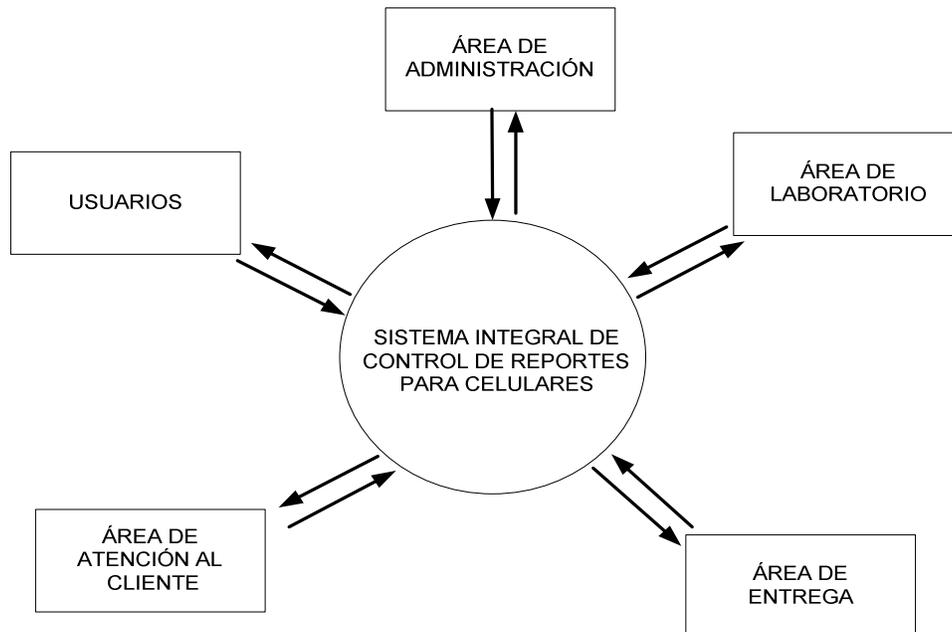


Figura IV.2.1.2. Diagrama de Contexto General del Sistema Nivel 0



En este diagrama se muestra un panorama general de las diferentes entidades que intervienen en el sistema, así como el flujo de información entre ellos, lo cual significa que nuestro sistema recibirá datos de las entidades, los procesará y emitirá resultados que involucren a otras entidades o a la misma que generó la información.

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

IV.2.2 Diagrama de flujo de procesos

Este diagrama permite conocer como será construido el sistema. Los componentes de un diagrama de flujo de procesos son los siguientes:

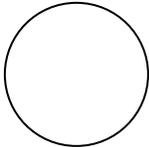
SIMBOLO	SIGNIFICADO
	Una entidad externa es un elemento del sistema, generalmente son personas, organizaciones, o sistemas de cómputo externos que pueden enviar o recibir datos del sistema.
	Un proceso se indica mediante un círculo, los procesos denotan un cambio o transformación de los datos.
	Un almacén de datos es representado por dos líneas paralelas. Se utilizan para almacenar datos.
	Un elemento de datos es representado por una flecha; la cabeza de la flecha indica la dirección del flujo de datos. Conecta a las entidades, procesos y almacenes.

Figura IV.2.2.1 Notación gráfica del diagrama de flujo de procesos



El diagrama de flujo de procesos general se muestra en la figura IV.2.2.2., como se observa en el diagrama se requiere que los usuarios del sistema se validen para poder acceder a el mediante un password, se verificará que dicha cuenta de usuario exista, haciendo una consulta a la base de datos.

Una vez validado se puede acceder a los diversos módulos que componen nuestro sistema:

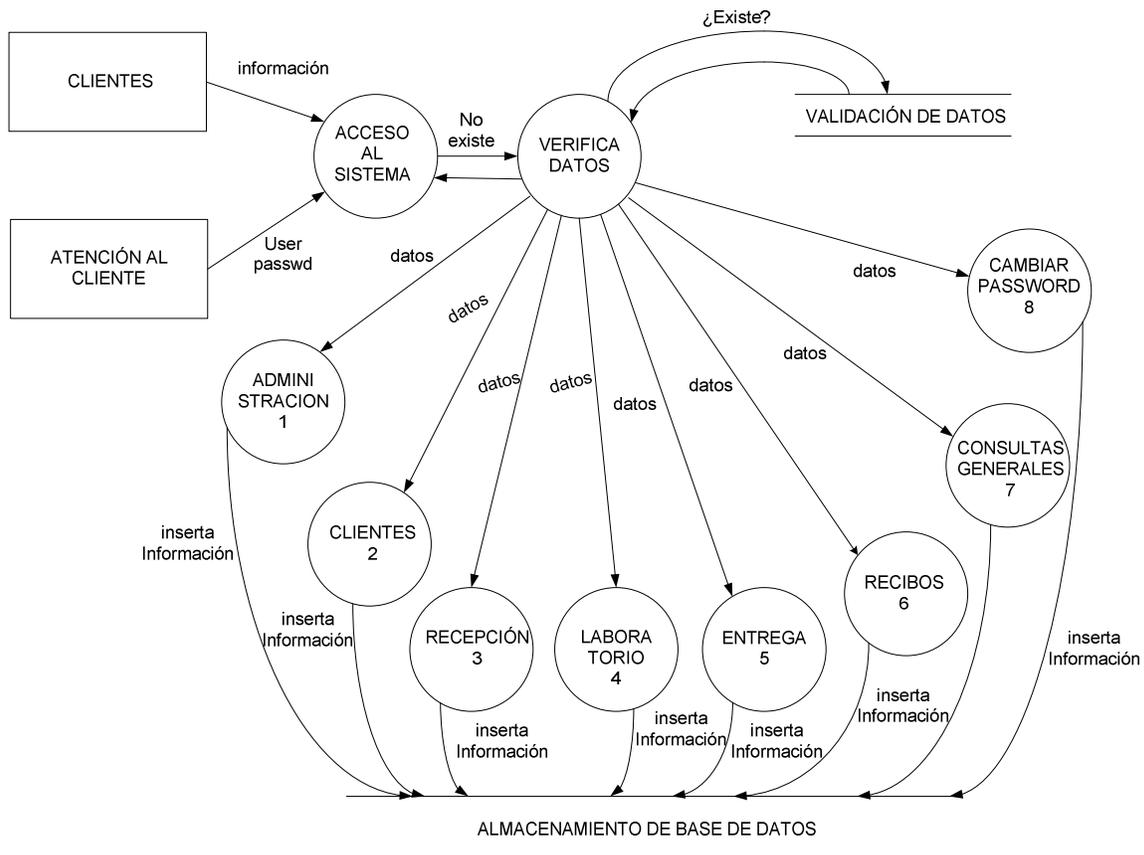


Figura IV.2.2.2 Diagrama de flujo de procesos nivel 1.

Módulo administración

Para el correcto análisis del sistema, se debe desglosar a detalle cada una de las burbujas o procesos, el primer módulo de administración se representa en la figura IV.2.2.3.

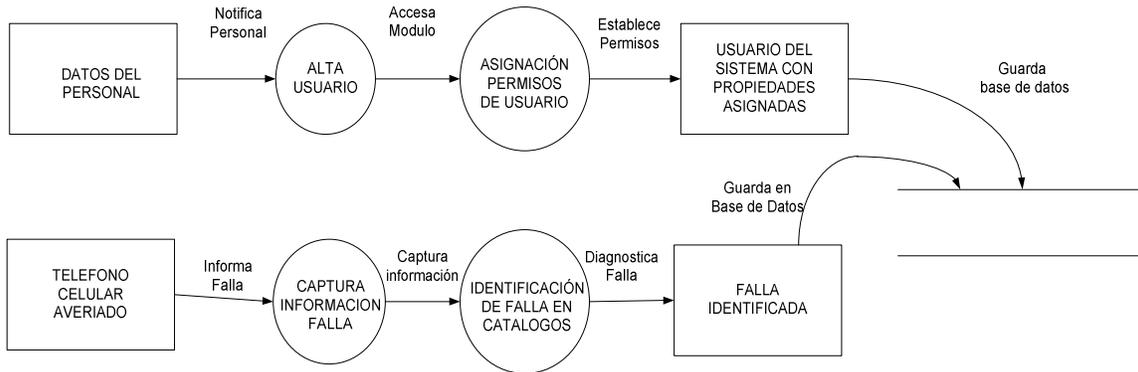


Figura IV.2.2.3 Diagrama de nivel 2 Módulo Administración.

Como se muestra en el diagrama anterior correspondiente al primer módulo de administración, primero se ingresan los datos del personal autorizado que hará uso del sistema, se le asignarán los permisos respectivos, los cuales serán almacenados en la base de datos. Un proceso similar se sigue al recibir el teléfono celular averiado, se recaba la información referente a la falla y se identifica dentro de los catálogos con los que se cuenta.

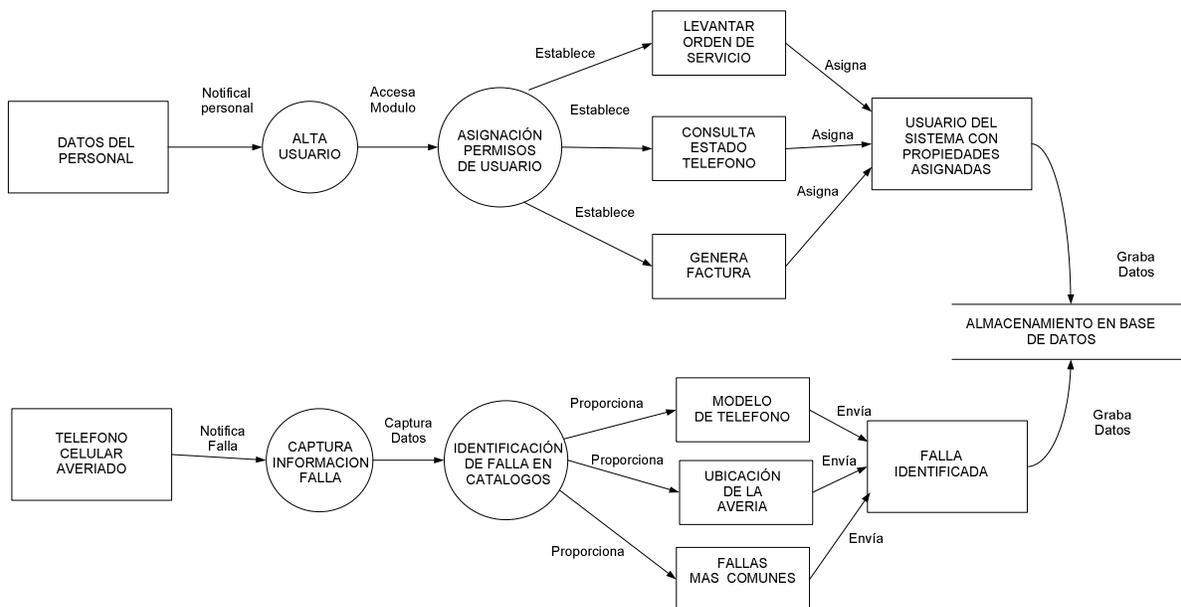


Figura IV.2.2.4 Diagrama de nivel 3 Módulo Administración.



Módulo clientes

Para dar de alta a un cliente se deben de insertar los datos requeridos por el sistema, como son el nombre, apellido, su dirección, teléfono y cualquier otro dato que sea de importancia. Una vez hecho esto se procede al almacenamiento de su información en la base de datos del sistema. Lo anterior se representa en las figuras IV.2.2.5 y IV.2.2.6.

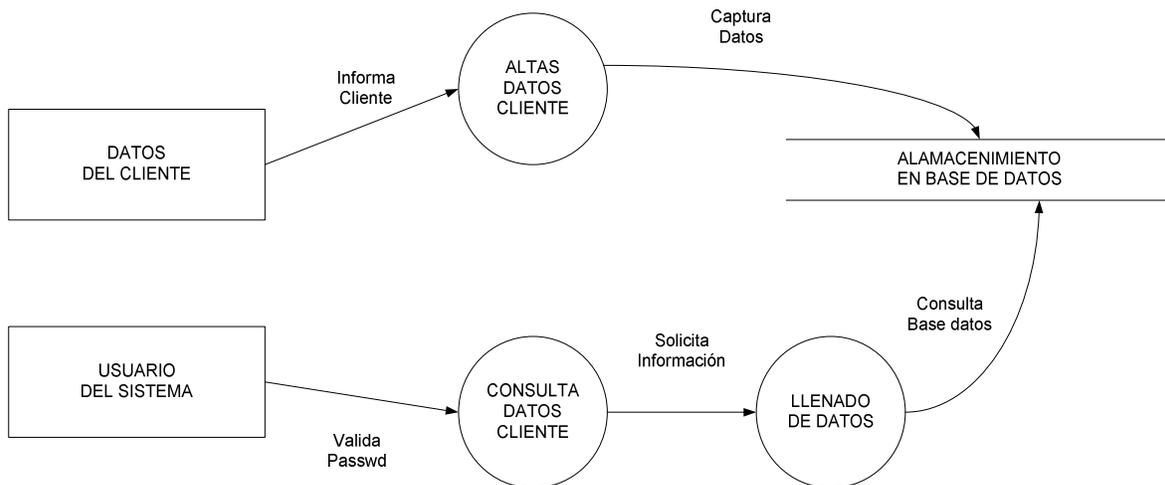


Figura IV.2.2.5 Diagrama de nivel 2 Módulo Clientes

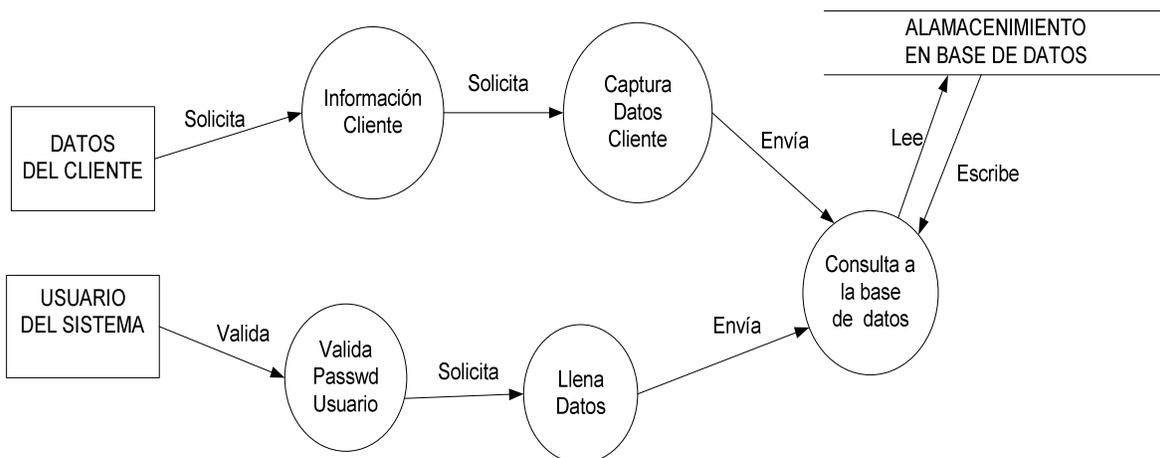


Figura IV.2.2.6 Diagrama de nivel 3 Módulo Clientes.



Módulo recepción

En este módulo se captura la información relativa al teléfono celular averiado, primeramente se dan de alta los datos del cliente, nombre y apellido, dirección, así también las características del teléfono celular (marca, modelo), una vez hecho esto la información se almacenará en la base de datos del sistema, el proceso se muestra en el diagrama de la figura IV.2.2.7 y el IV.2.2.8 respectivamente.

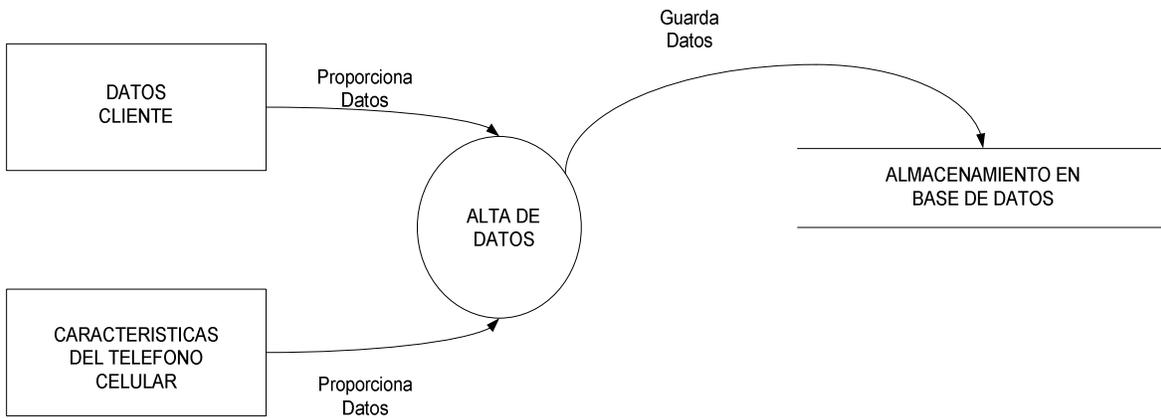


Figura IV.2.2.7 Diagrama de nivel 2 Módulo Recepción.

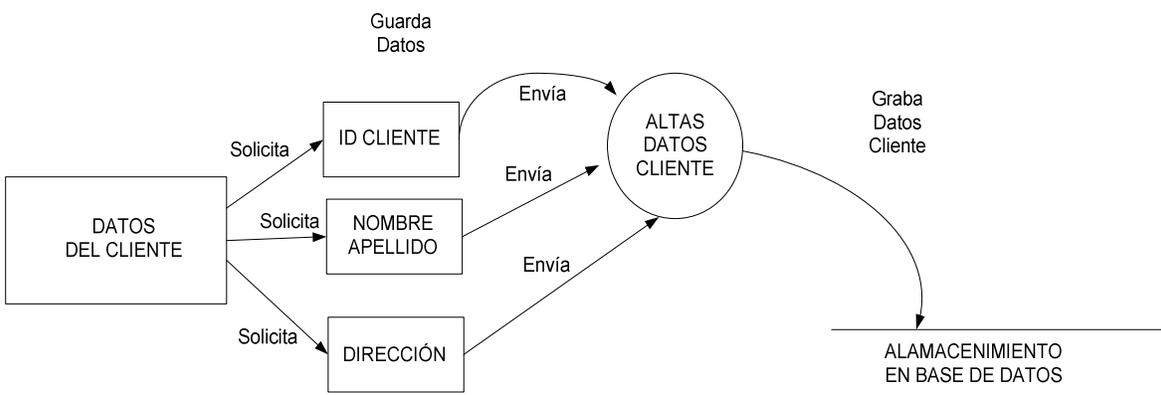


Figura IV.2.2.8 Diagrama de nivel 3 Módulo Recepción



Módulo de laboratorio

La información referente a los datos necesarios para el levantamiento de la orden de servicio y su posterior envío al técnico calificado, se muestra en el diagrama de la figura IV.2.2.9 y IV.2.2.10.

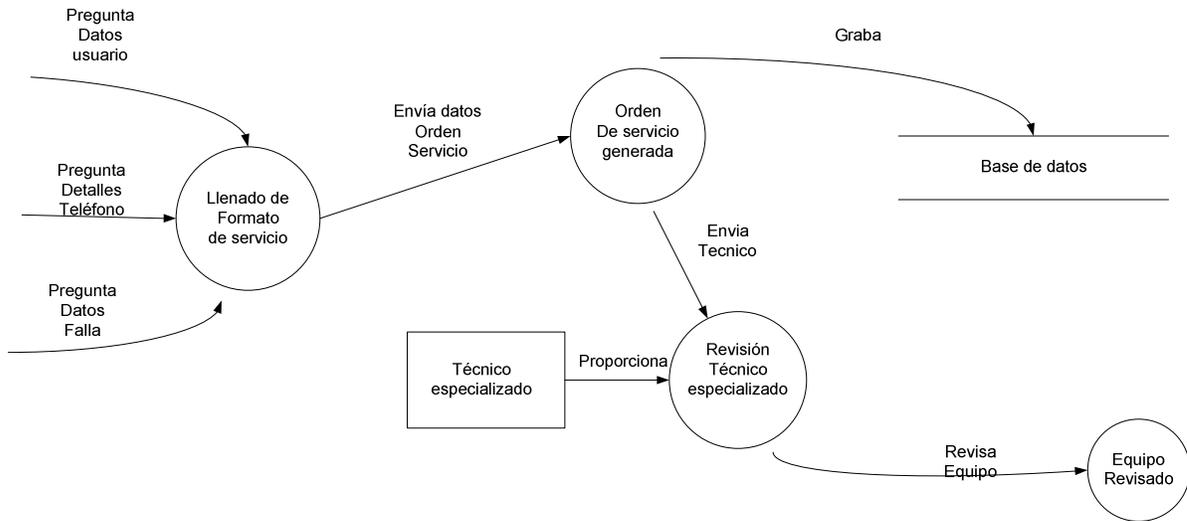


Figura IV.2.2.9 Diagrama de Nivel 2 Módulo Laboratorio.

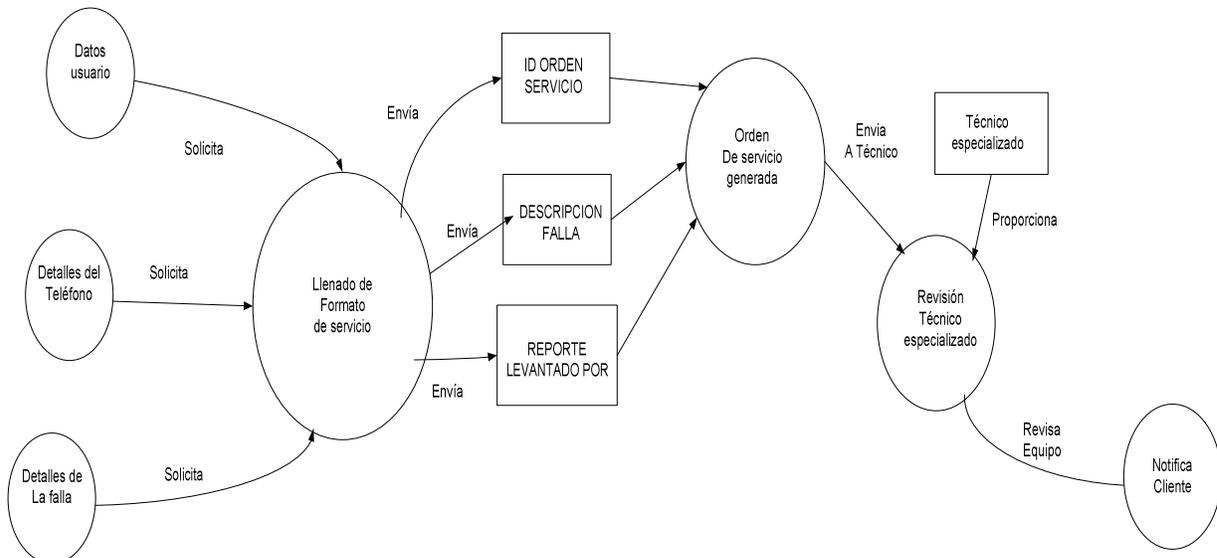


Figura IV.2.2.10 Diagrama de Nivel 3 Módulo Laboratorio



Módulo de entrega

El proceso de entrega del teléfono celular se inicia cuando con los datos del cliente, el número de la orden de servicio se procesa la orden de salida, una vez hecho esto se le notifica al usuario que su equipo se ha reparado, el cual es enviado al área de entregas para que pueda recogerlo, como se muestra en el diagrama IV.2.2.11 y IV.2.2.12 .

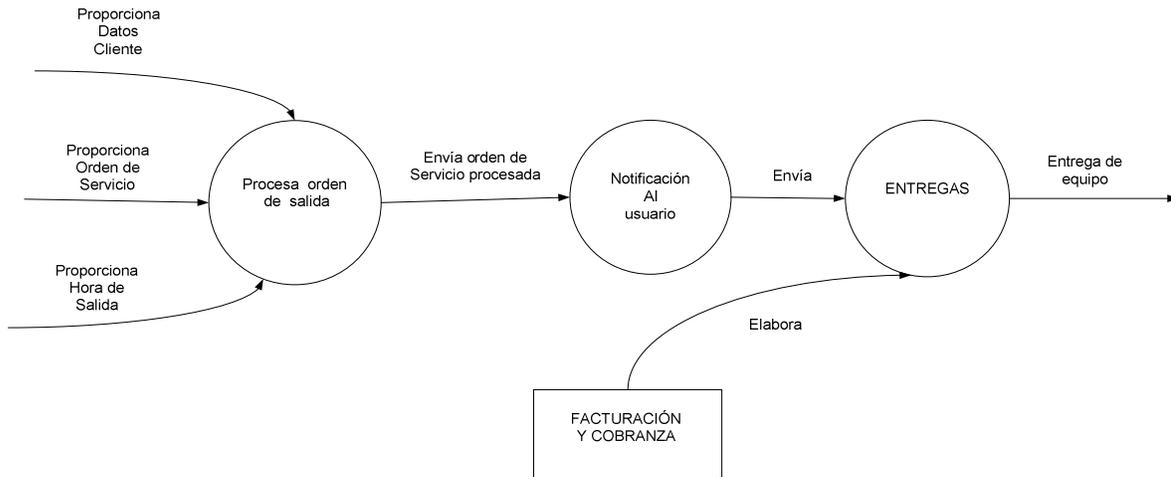


Figura IV.2.2.11. Diagrama nivel 2 Módulo Entrega.

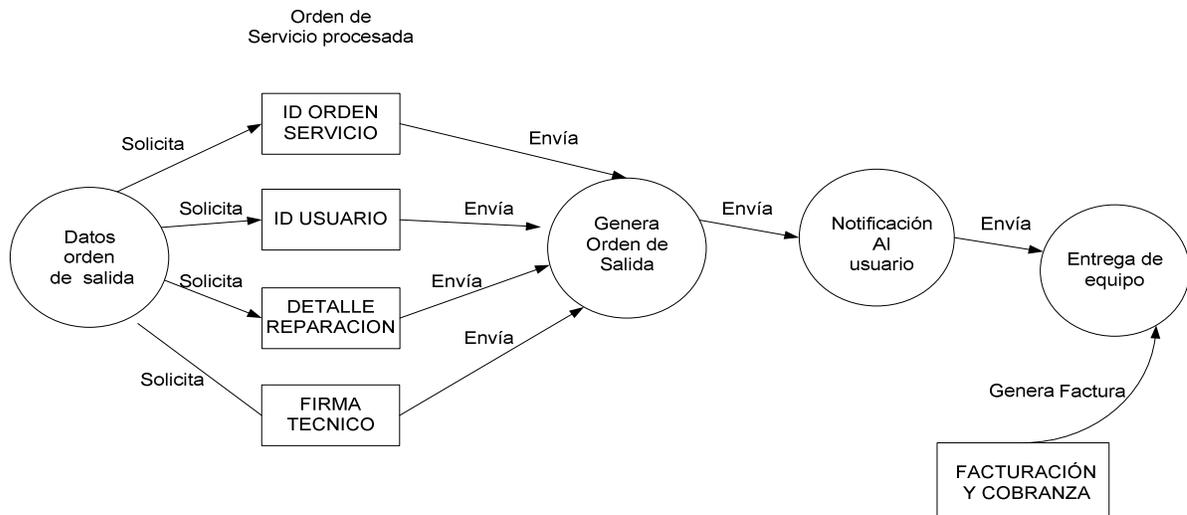


Figura IV.2.2.12. Diagrama nivel 3 Módulo Entrega.



Módulo de recibos

Para la expedición de una factura o un recibo, se sigue el proceso mostrado en el diagrama de la figura IV.2.2.13 y IV.2.2.14.

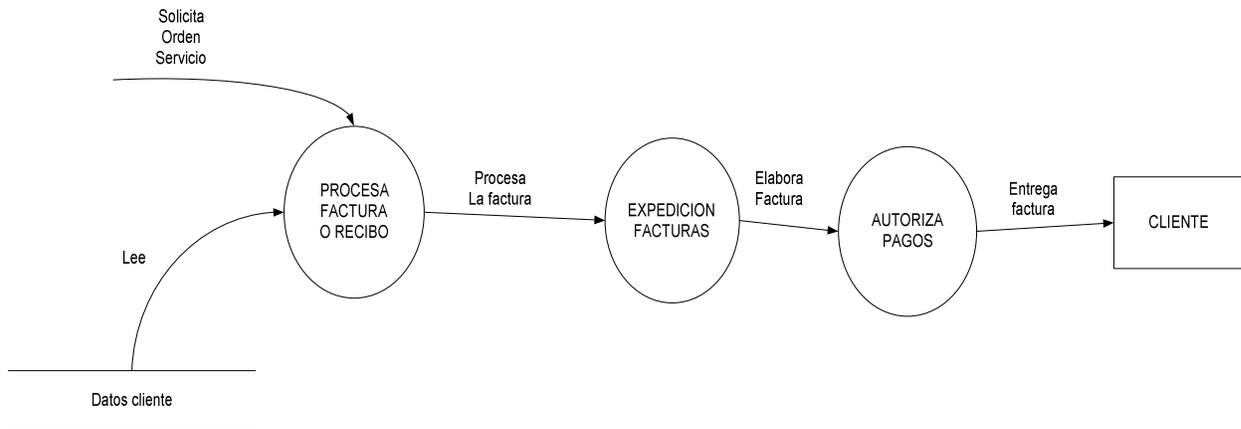


Figura IV.2.2.13 Diagrama nivel 2 Módulo Recibos

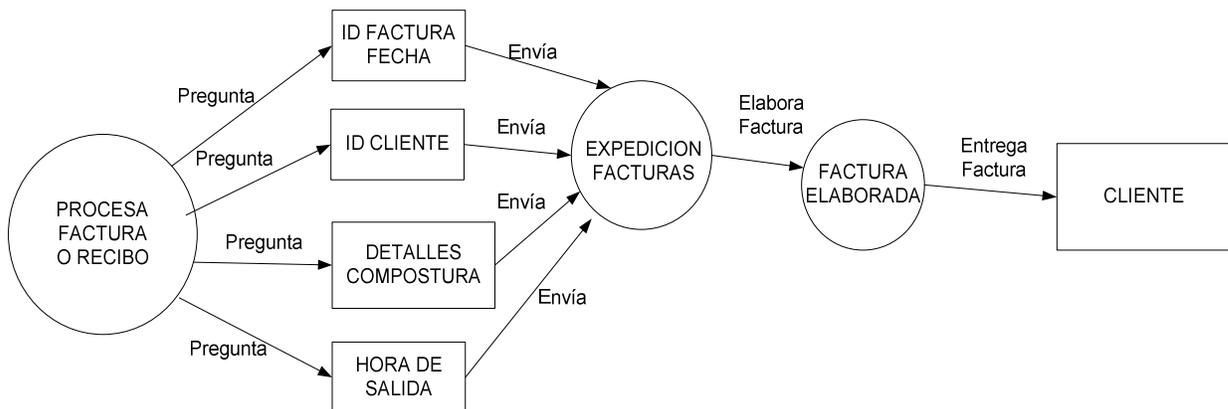


Figura IV.2.2.14 Diagrama Nivel 3 Módulo Recibos

Módulo de consulta general

Para cualquier consulta que se desee realizar al sistema se cuenta con un módulo que nos permite solicitar información respecto al estado en el que se



encuentra la orden del servicio, entre otras funciones. A continuación en el diagrama de la figura IV.2.2.15 y IV.2.2.16 se muestra el proceso que se sigue.

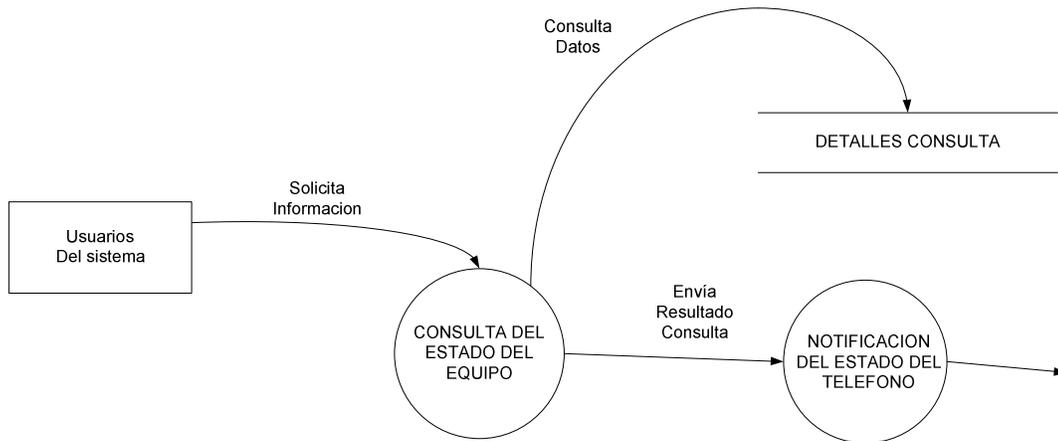


Figura IV.2.2.15. Diagrama Nivel 2 Módulo Consulta General.

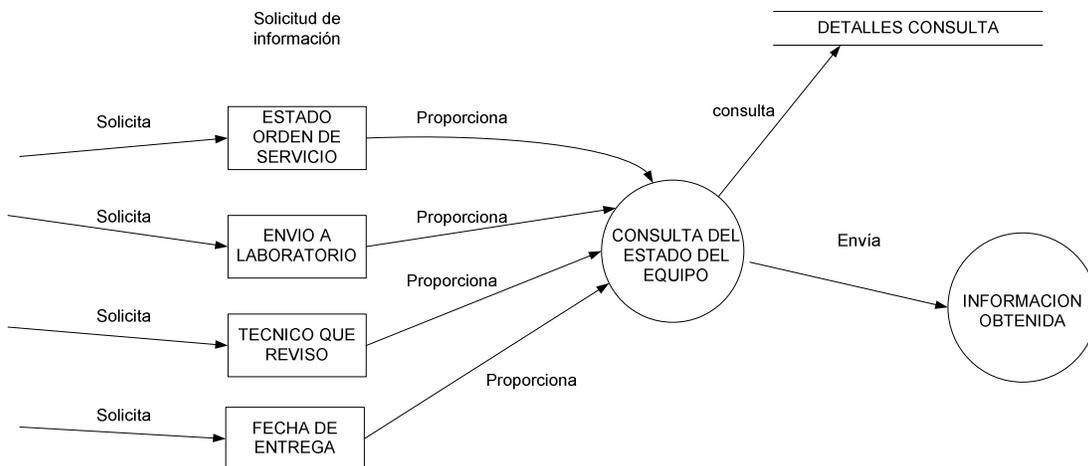


Figura IV.2.2.16. Diagrama Nivel 3 Módulo Consulta General.



Módulo de cambio de password

Para modificar la contraseña o el password de algún usuario del sistema, se requiere primeramente que se proporcione el anterior password y el nuevo, con la finalidad de que sea mas seguro para el usuario. La nueva contraseña se almacenará en la base de datos. Se muestra el proceso en la figura IV.2.2.17 y IV.2.2.18.

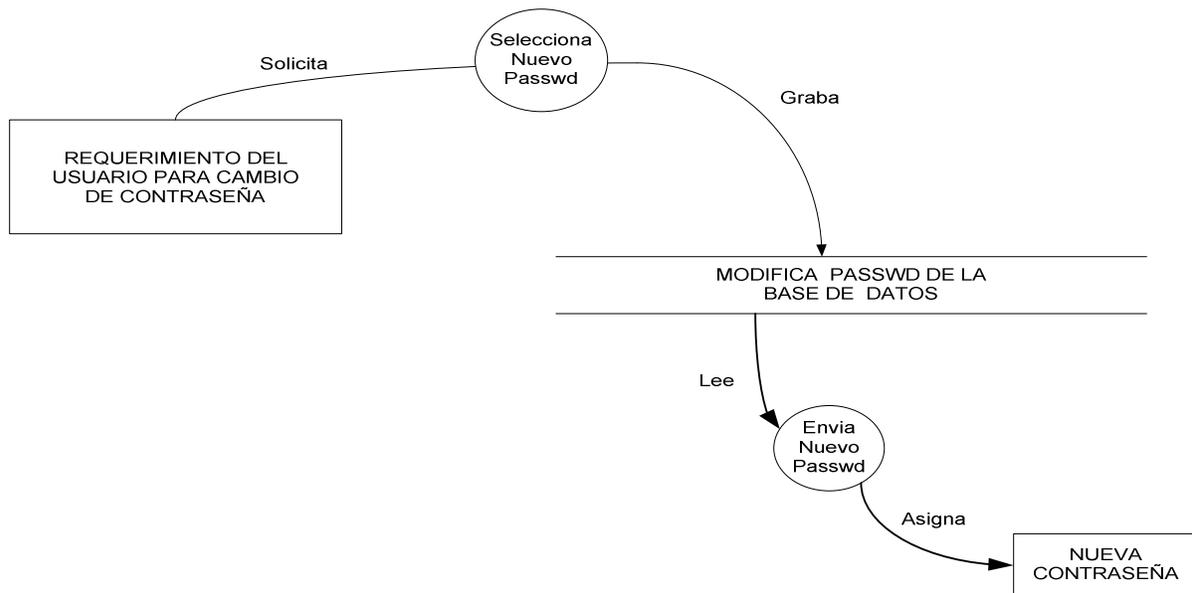


Figura IV.2.2.17 Diagrama nivel 2 Módulo Cambio de Password.

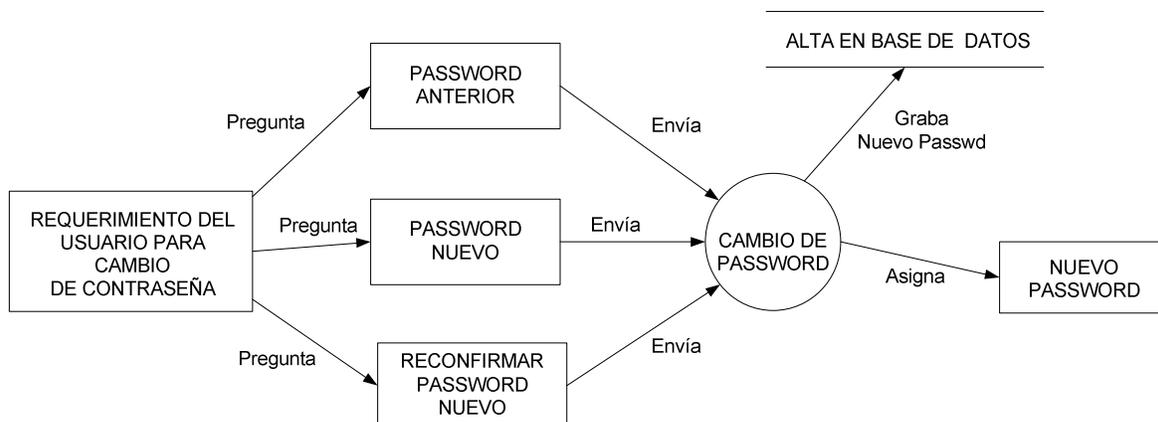


Figura IV.2.2.18. Diagrama Nivel 3 Módulo Cambio de Password.



IV.2.3 DFD general

Debido a que el diagrama de flujo de procesos no maneja con detalle las operaciones que deberá realizar el sistema y que acciones deberá involucrar, se utilizará el DFD para poder enfocarse en la organización y en la forma en la que opera el sistema, ver figuras IV.2.3.1., IV.2.3.2., IV.2.3.3., IV.2.3.4. y IV.2.3.5. Etc. Etc.

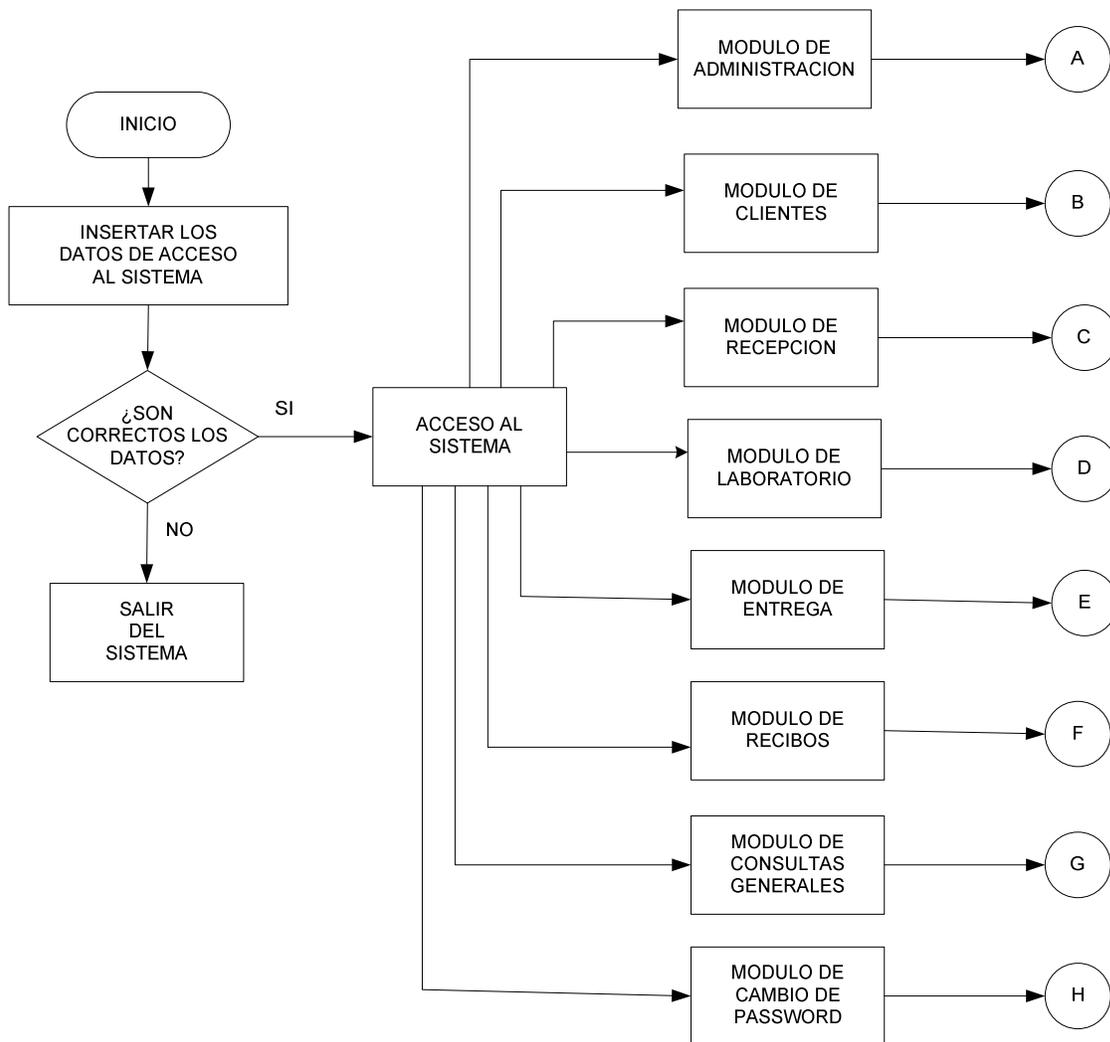


Figura IV.2.3.1 DFD general.

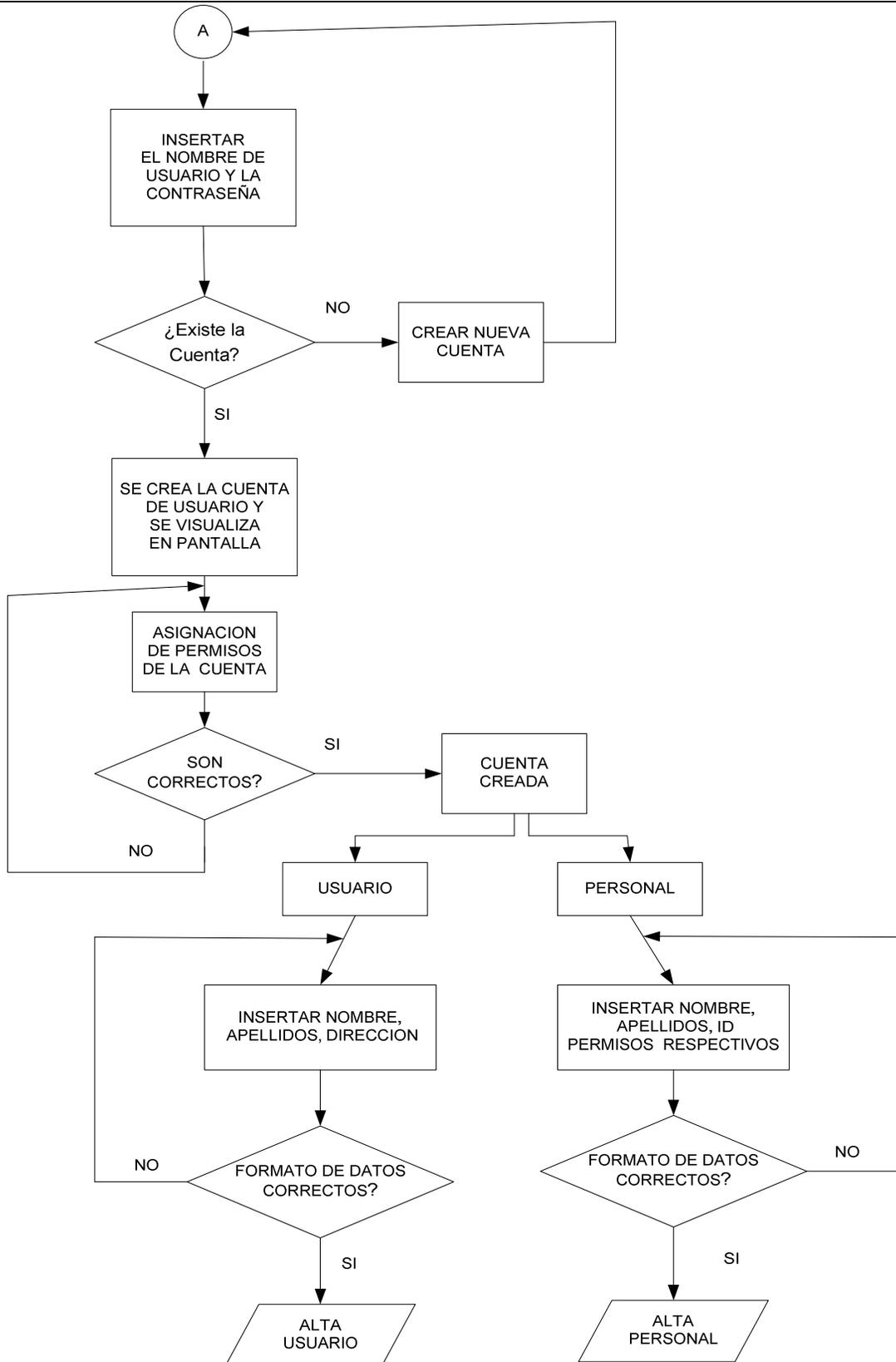


Figura IV.2.3.2 DFD Módulo A Administración

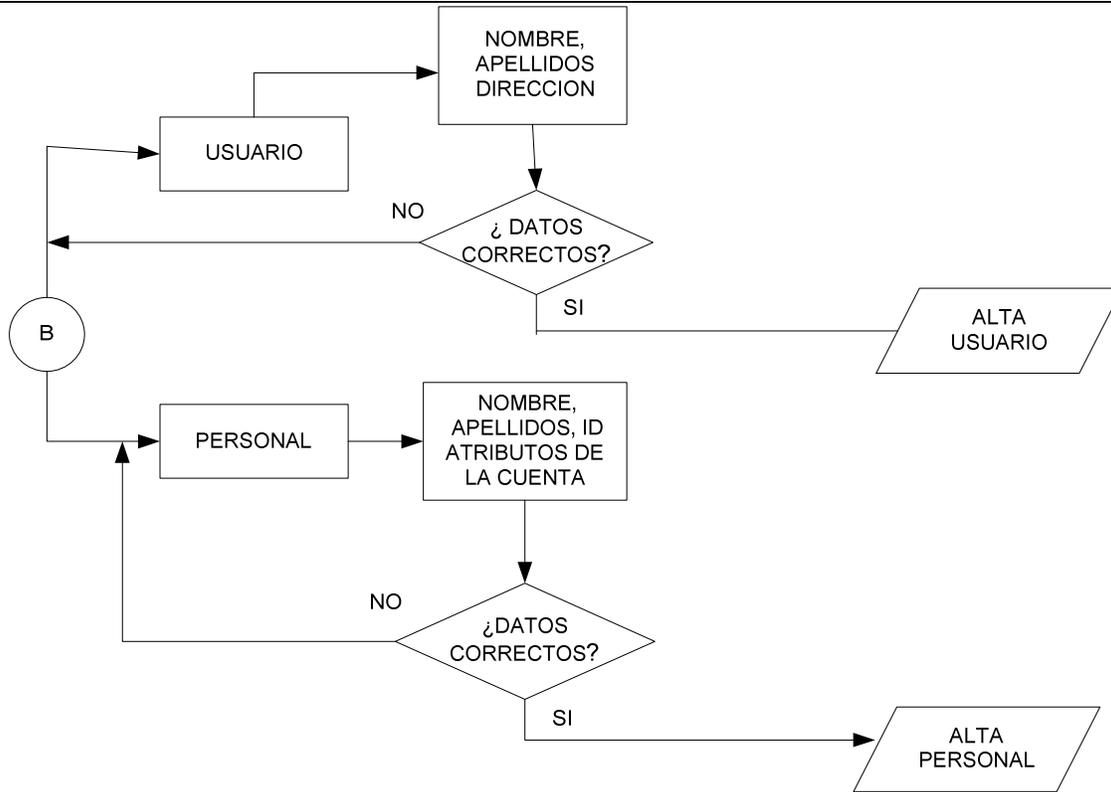


Figura IV.2.3.3 DFD Módulo B Clientes

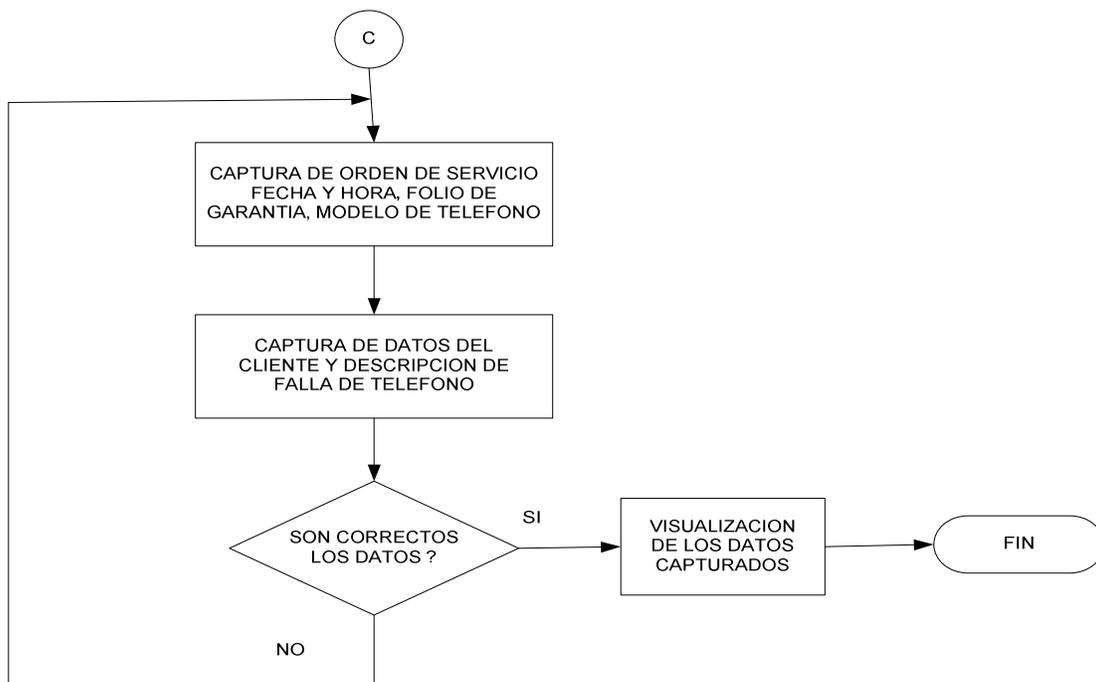


Figura IV.2.3.4 DFD Módulo C Recepción

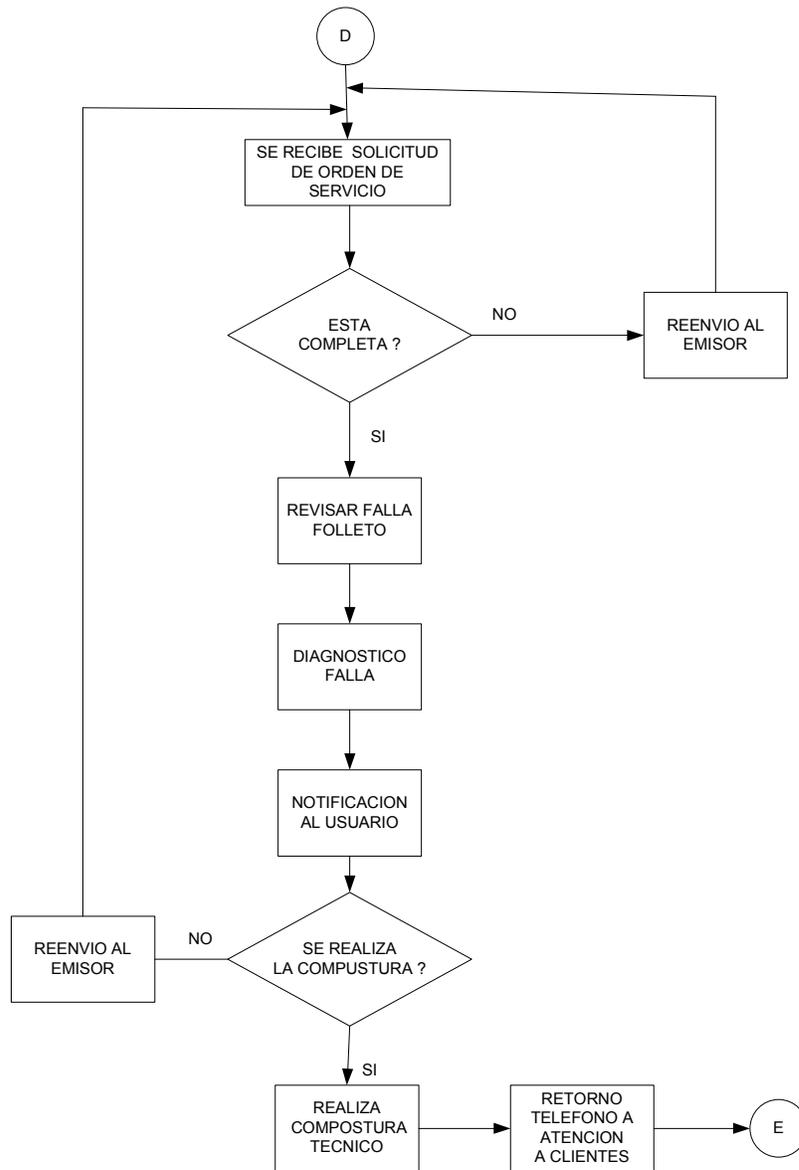


Figura IV.2.3.5 DFD Módulo D Laboratorio.

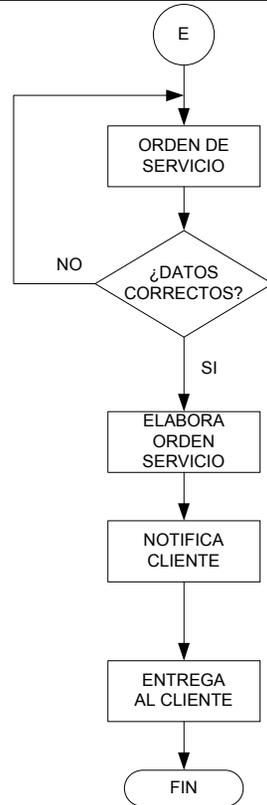


Figura IV.2.3.6 DFD Módulo E Entregas.

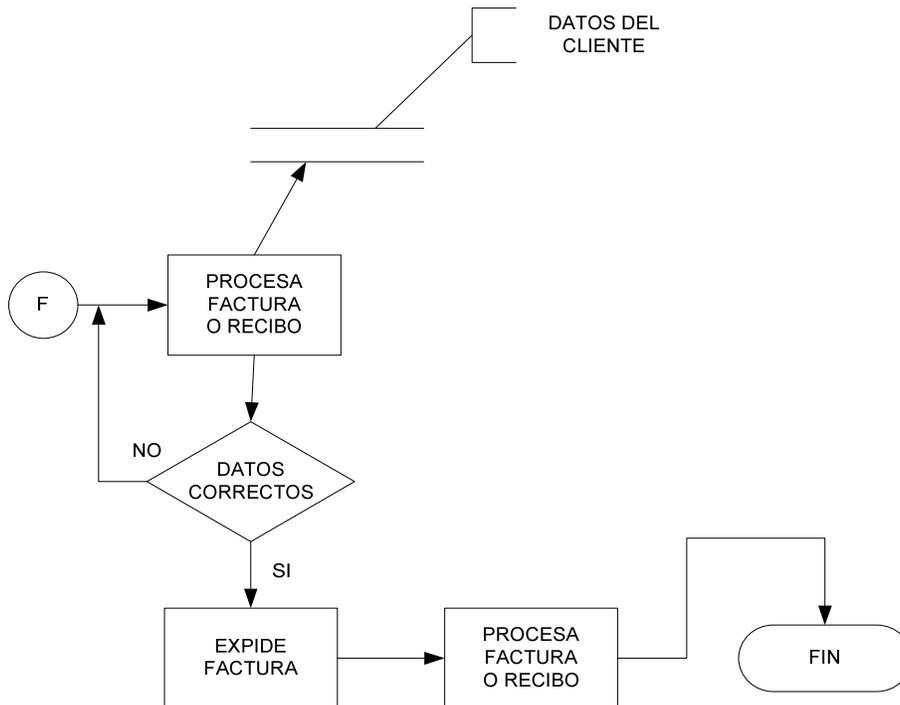


Figura IV.2.3.7 DFD Módulo F Recibos.

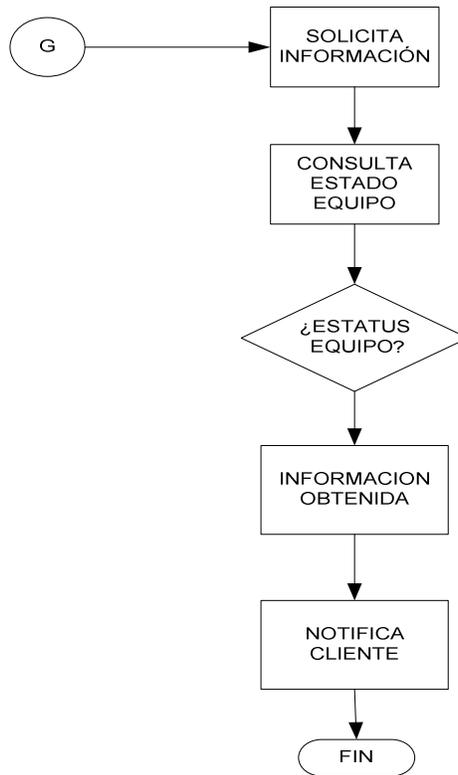


Figura IV.2.3.8 DFD Módulo G Consultas Generales



Figura IV.2.3.9 DFD módulo H Cambio de Password



IV.2.4 Diccionario de Datos

El Diccionario de Datos es una forma de documentación para el diseñador de bases de datos, su utilidad básica se describe en las siguientes funciones:

- Son datos comúnmente llamados metadatos.
- Describir el propósito de las Bases de Datos
- Proveer una detallada descripción de cada tabla dentro de la base de datos y explicar su propósito
- Documentar la estructura interna de cada tabla, incluyendo todos sus campos, sus tipos de datos, todos los índices y todas las vistas.
- Describir reglas como pueden ser el uso de valores nulos, valores únicos.

Las tablas que conforman el diccionario de datos se mencionan a continuación: En la tabla cliente (ver tabla IV.2.4.1) se pide información específica del cliente para ser almacenada dentro de la base de datos. La tabla IV.2.4.2 representa la tabla usuarios, en esta tabla los campos requeridos sirven para identificar al personal que labora en la empresa y asignar los permisos del personal en su perfil. La tabla IV.2.4.3 representa la incidencia del equipo del cliente y sirve para relacionar las tablas clientes y usuarios.

En la tabla personal (ver tabla IV.2.4.4) se ingresa información del empleado. La tabla recepción incidencias (ver tabla IV.2.4.5) contiene información de las causas por las que ha ingresado al laboratorio en determinado momento, el equipo del cliente. La tabla IV.2.4.6 incluye los campos donde se describe el modelo del teléfono, la tecnología usada por el equipo entre otros. En la tabla IV.2.4.7 se muestra la tabla proveedor de servicio la cual se relaciona con la tabla recepción (ver tabla IV.2.4.8), en esta tabla, se recaba la información de las condiciones en las que ingresa el equipo.



Para la tabla fallas (ver tabla IV.2.4.9) se describe el código de la falla y la descripción de está. Por otra parte la tabla Referencia de proveedor (ver tabla IV.2.4.10) y la tabla pertenece a (ver tabla IV.2.4.11) se relacionan con la tabla recepción (ver tabla IV.2.4.8).

Las tablas laboratorio incidencias (ver tabla IV.2.4.12), tabla códigos de transacción (ver tabla IV.2.4.13) y tabla problema encontrado (ver tabla IV.2.4.14) están relacionadas con la tabla laboratorio (ver tabla IV.2.4.15). En estas tablas se recopila la información sobre los problemas que son detectados en el equipo telefónico. También se relacionan las tablas laboratorio detalles (ver tabla IV.2.4.16), tabla códigos de reparación (ver tabla IV.2.4.17), tabla reparar estatus código (ver tabla IV.2.4.18) con la tabla laboratorio y en las cuales se manejan los códigos de reparación del equipo celular.

Las tablas código de partes (ver tabla IV.2.4.19), código de partes dañadas (ver tabla IV.2.4.20) y tabla código de referencia (ver tabla IV.2.4.21) contienen los códigos y descripciones de las fallas del aparato. Finalmente la tabla entrega (ver tabla IV.2.4.22) contiene la información para la entrega del equipo a su dueño. La tabla entrega incidencias (ver tabla IV.2.4.23) se relaciona con la tabla entrega y la tabla usuarios.

Tabla Clientes (tab_clientes)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Identificador del cliente	idcliente	Integer (4)	SI	NO	NOT NULL	Id. único del cliente
Nombre del Cliente	cliente	Varchar (50)	NO	NO	NOT NULL	Contiene el nombre del cliente
Nombre de la Empresa	empresa	Varchar (50)	NO	NO	NULL	Contiene el nombre de la empresa
Calle	calle_num	Varchar (50)	NO	NO	NULL	Contiene la calle y el número del cliente



Análisis, Diseño y Construcción del Sistema

Colonia	colonia	Varchar (50)	NO	NO	NULL	Contiene la colonia del cliente
Delegación / Municipio	del_municipio	Varchar (50)	NO	NO	NULL	Contiene la delegación o municipio del cliente
Estado	estado	Varchar (50)	NO	NO	NULL	Contiene el estado de residencia del cliente
Código Postal	cp	Integer (6)	NO	NO	NULL	Contiene el código postal del domicilio del cliente
Teléfono	tel	Integer (20)	NO	NO	NULL	Contiene el número telefónico del cliente
Mail	mail	Varchar (30)	NO	NO	NULL	Contiene el correo electrónico del cliente
Registro Federal de Causante	rfc	Integer (13)	NO	NO	NULL	Contiene el RFC del cliente
Fecha de Registro	fecha_registro	datetime (8)	NO	NO	NULL	Muestra la fecha de registro en el sistema

Tablas con las que se relaciona tab_clientes

Tablas: tab_clientes incidencias, tab_recepcion

Tabla IV.2.4.1 Tabla Clientes

Tabla Usuarios (tab_usuarios)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Identificador del personal	ldpersonal	Integer (7)	NO	SI	NOT NULL	Identificador único del personal que administra la BD
Nombre del personal	Usuario	Varchar (10)	SI	NO	NOT NULL	Nombre del personal que administra la BD
Contraseña	Contraseña	Varchar (10)	NO	NO	NOT NULL	Contraseña de acceso al sistema
Vista Administración	v_administracion	Char (1)	NO	NO	NULL	Formulario de Admón.



Análisis, Diseño y Construcción del Sistema

Vista Clientes Alta	v_clientes_alta	Char (1)	NO	NO	NULL	Formulario de Alta de Cliente
Vista Clientes Consulta	v_clientes_consulta	Char (1)	NO	NO	NULL	Formulario de Consulta de Clientes
Vista Recepción	v_recepcion	Char (1)	NO	NO	NULL	Formulario Recepción
Vista Recepción Captura	v_recepcion_captura	Char (1)	NO	NO	NULL	Formulario Captura de Recepción
Vista Recepción Cambios	v_recepcion_cambios	Char (1)	NO	NO	NULL	Formulario Cambios en Recepción
Vista Recepción Consultas	v_recepcion_consultas	Char (1)	NO	NO	NULL	Formulario Consulta de Recepciones
Vista Laboratorio	v_laboratorio	Char (1)	NO	NO	NULL	Formulario Laboratorio
Vista Laboratorio Captura	v_laboratorio_captura	Char (1)	NO	NO	NULL	Formulario Captura de Laboratorio
Vista Laboratorio Cambios	v_laboratorio_cambios	Char (1)	NO	NO	NULL	Formulario Cambios en Laboratorio
Vista Laboratorio Consultas	v_laboratorio_consultas	Char (1)	NO	NO	NULL	Formulario Consultas Laboratorio
Vista Entrega	v_entrega	Char (1)	NO	NO	NULL	Formulario de Entrega
Vista Entrega Captura	v_entrega_captura	Char (1)	NO	NO	NULL	Formulario Captura de Entrega
Vista Entrega Cambios	v_entrega_cambios	Char (1)	NO	NO	NULL	Formulario Cambios de Entrega
Vista Entrega Consultas	v_entrega_consultas	Char (1)	NO	NO	NULL	Formulario Consulta de Entrega
Vista Recibos	v_recibos	Char (1)	NO	NO	NULL	Formulario de Recibos
Vista Consultas Generales	v_consultas_generales	Char (1)	NO	NO	NULL	Formulario Consultas Generales

Tablas con las que se relaciona tab_usuarios

Tablas: tab_clientes_incidentes, tab_recepcion_incidentes, tab_laboratorio_incidentes, tab_entrega_incidentes

Tabla IV.2.4.2 Tabla Usuarios



Tabla Clientes Incidencias (tab_clientes_incidencias)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Incidencia del Equipo	Incidencia	Integer (5)	NO	NO	NULL	Causas de ingreso del equipo
Identificador del Cliente	Idcliente	Integer (4)	NO	SI	NOT NULL	Id. único del cliente
Nombre del Personal	Usuario	Varchar (10)	NO	SI	NOT NULL	Nombre del personal que administra la BD
Fecha de Incidencia	Fecha	Datetime (8)	NO	NO	NULL	Contiene la fecha de incidencia

Tablas con las que se relaciona tab_clientes_incidencias

Tablas: tab_usuarios, tab_clientes

Tabla IV.2.4.3 Tabla Clientes Incidencias

Tabla Personal (tab_personal)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Identificador del empleado	Idpersonal	Integer (7)	SI	NO	NOT NULL	Identificador único del empleado
Nombre del empleado	Nombre	Varchar (50)	NO	NO	NOT NULL	Nombre del empleado
Dirección	Direccion	Varchar (50)	NO	NO	NULL	Domicilio del empleado
Teléfono	Telefono	Integer (20)	NO	NO	NULL	Número de teléfono del empleado
Celular	Movil	Integer (20)	NO	NO	NULL	Número de celular del empleado

Tablas con las que se relaciona tab_personal

Tablas: tab_usuarios

Tabla IV.2.4.4 Tabla Personal

Tabla Recepción Incidencias (tab_recepcion_incidencias)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Incidencia del equipo	Incidencia	Integer (5)	NO	NO	NULL	Causas de ingreso del equipo
Año	año	Integer (4)	NO	NO	NULL	Año en que ingreso el equipo
Orden de servicio	orden_servicio	Integer (5)	NO	SI	NOT NULL	Número de orden de servicio



Nombre del personal	usuario	Varchar (10)	NO	SI	NOT NULL	Nombre del personal que administra la BD
Fecha de incidencia	fecha	Datetime (8)	NO	NO	NULL	Contiene la fecha de incidencia del equipo

Tablas con las que se relaciona tab_recepcion_incidencias

Tablas: tab_recepcion, tab_usuarios

Tabla IV.2.4.5 Tabla Recepción Incidencias

Tabla Modelo Equipos (tab_modeloEquipos)

Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Modelo de teléfono	modelo_tel	Varchar (20)	SI	NO	NOT NULL	Contiene el modelo del equipo celular
Tecnología	tecnologia	Varchar (10)	NO	NO	NULL	Tecnología que utiliza el equipo celular
Marca	marca	Varchar (20)	NO	NO	NULL	Marca del equipo celular
Nombre del personal	usuario	Varchar (10)	NO	NO	NULL	Nombre del personal que administra la BD

Tablas con las que se relaciona tab_modeloEquipos

Tablas: tab_recepcion

Tabla IV.2.4.6 Tabla Modelo Equipos

Tabla Proveedor de Servicio (tab_carrier_service_provider)

Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Código del carrier	cod_carrier	Varchar (8)	SI	NO	NOT NULL	Código del proveedor
Descripción	descripcion	Varchar (50)	NO	NO	NULL	Descripción del proveedor
País	pais	Varchar (50)	NO	NO	NULL	Contiene el país del proveedor

Tablas con las que se relaciona tab_carrier_service_provider

Tablas: tab_recepcion

Tabla IV.2.4.7 Tabla Proveedor de Servicio

Tabla Recepción (tab_recepcion)

Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Año de ingreso	año	Integer (4)	NO	NO	NULL	Año en que ingreso el equipo celular
Orden de Servicio	orden_servicio	Integer (5)	SI	NO	NOT NULL	Número de orden de servicio
Folio de garantía	folio_garantia	Integer (10)	NO	NO	NULL	Folio de garantía del equipo celular



Análisis, Diseño y Construcción del Sistema

Fecha	fecha	Datetime (8)	NO	NO	NOT NULL	Fecha de recepción del equipo celular
Hora	hora	Datetime (5)	NO	NO	NOT NULL	Hora de recepción del equipo celular
Modelo del teléfono	modelo_tel	Varchar (20)	NO	SI	NOT NULL	Contiene el modelo del equipo celular
imei	imei	Integer (15)	NO	NO	NOT NULL	imei del equipo celular
msn	msn	Varchar (10)	NO	NO	NULL	msn del equipo celular
esc-csn	esn_csn	Varchar (11)	NO	NO	NOT NULL	esc-csn del equipo celular
Código del tranceptor	cod_tranceptor	Integer (20)	NO	NO	NOT NULL	Código de tranceptor del equipo celular
Código apc	cod_apc	Integer (20)	NO	NO	NOT NULL	Código apc del equipo celular
Código de fabricación	cod_fabricacion	Integer (20)	NO	NO	NULL	Código de fabricación del equipo celular
Identificador de origen	id_origen	Integer (20)	NO	NO	NULL	Identificador de origen del equipo
Código del proveedor	cod_carrier	Varchar (3)	NO	SI	NOT NULL	Código del proveedor
Versión de software	ver_sw_entrada	Varchar (20)	NO	NO	NULL	Versión software de entrada del equipo celular
Numero de minutos	no_min	Integer (10)	NO	NO	NULL	Número de minutos en el equipo celular
Numero de minutos acumulados	no_min_acum	Integer (9)	NO	NO	NULL	Numero de minutos acumulados en el equipo celular
Tarjeta sim	tarjeta_sim	Char (1)	NO	NO	NULL	Tarjeta del equipo celular
Número de guía entrada	no_guia_entrada	Varchar (20)	NO	NO	NOT NULL	Número de guía de entrada
Goma	goma	Char (1)	NO	NO	NULL	Goma del celular
Cargador	cargador	Char (1)	NO	NO	NULL	Cargador del equipo celular
Batería	bateria	Char (1)	NO	NO	NULL	Batería del celular
Antena	antena	Char (1)	NO	NO	NULL	Antena del celular
Tapa	tapa	Char (1)	NO	NO	NULL	Tapa del celular
Hotler	hotler	Char (1)	NO	NO	NULL	Hotler del celular



Análisis, Diseño y Construcción del Sistema

Número de cim	no_cim	Integer (20)	NO	NO	NULL	Número de cim del equipo
Garantía de la factura	garantia_factura	Char (1)	NO	NO	NULL	Garantía de factura del equipo
Fecha de facturación	fecha_factura	Datetime (8)	NO	NO	NULL	Fecha de la factura del equipo
Código apc de accesorio	cod_apc_accesorio	Varchar (20)	NO	NO	NULL	Código de apc de accesorio del equipo celular
Código de falla	codigo_falla	Varchar (5)	NO	SI	NOT NULL	Código de falla del equipo celular
Garantía	garantia	Char (1)	NO	NO	NULL	Garantía del equipo celular
Costo	costo	Char (1)	NO	NO	NULL	Costo de la reparación del equipo celular
Valor del equipo	valor	Numeric (10)	NO	NO	NULL	Costo del equipo celular
Numero de reincidencias	no_reincidencias	Integer (5)	NO	NO	NULL	Número de veces que ha ingresado el equipo celular
Observaciones	observaciones	Varchar (50)	NO	NO	NULL	Observaciones que considere el personal
Recibido	recibido_por	Varchar (50)	NO	NO	NOT NULL	Persona que recibe el celular
Identificador del Cliente	id_cliente	Integer (4)	NO	SI	NOT NULL	Identificador único del cliente
Status	status	Varchar (50)	NO	NO	NULL	Status del equipo celular
Memoria	memoria	Char (1)	NO	NO	NULL	Memoria del equipo celular
Capacidad de Memoria del Equipo	memoria_capacidad	Varchar (5)	NO	NO	NULL	Capacidad de memoria del equipo celular
Código de referencia del carrier	cod_ref_carrier	Varchar (3)	NO	SI	NOT NULL	Código que maneja el proveedor
Tapón	tapon	Char (1)	NO	NO	NULL	Tapón del equipo celular
Código Pertenece a	cod_pertenece_a	Varchar (50)	NO	SI	NOT NULL	Código pertenece a, manejado por el proveedor
Cotización	cotizar	Char (1)	NO	NO	NULL	Cotización del equipo celular
Manos Libres	manos_libres	Char (1)	NO	NO	NULL	Manos libres del equipo celular
Interfaz	interfaz	Char (1)	NO	NO	NULL	Interfaz del equipo celular



Bluetooth	bluetooth	Char (1)	NO	NO	NULL	Tecnología disponible en el equipo celular
-----------	-----------	----------	----	----	------	--

Tablas con las que se relaciona tab_recepcion

Tablas: tab_modelo_equipo, tab_carrier_services_provider, tab_fallas, tab_referencia_carrier, tab_pertenece_a, tab_recepcion_incidencias

Tabla IV.2.4.8 Tabla Recepción

Tabla Fallas (tab_fallas)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Código de falla	codigo_falla	Varchar (5)	SI	NO	NOT NULL	Código de falla del equipo celular
Descripción	Descripción	Varchar (50)	NO	NO	NULL	Descripción de la falla del equipo

Tablas con las que se relaciona tab_fallas

Tablas: tab_recepcion

Tabla IV.2.4.9 Tabla Fallas

Tabla Referencia de Proveedor (tab_referencia_carrier)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Código de referencia del carrier	cod_ref_carrier	Varchar (3)	SI	NO	NOT NULL	Código que maneja el proveedor
Descripción	Descripción	Varchar (50)	NO	NO	NULL	Descripción del proveedor

Tablas con las que se relaciona tab_referencia_carrier

Tablas: tab_recepcion

Tabla IV.2.4.10 Tabla Referencia de Proveedor

Tabla Pertenece a (tab_pertenece_a)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Código Pertenece a	cod_pertenece_a	Varchar (50)	SI	NO	NOT NULL	Código pertenece a, manejado por el proveedor
Descripción	descripcion	Varchar (50)	NO	NO	NULL	Descripción del proveedor

Tablas con las que se relaciona tab_pertenece_a

Tablas: tab_recepcion

Tabla IV.2.4.11 Tabla Pertenece a



Tabla Laboratorio Incidencias (tab_laboratorio_incidencias)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Incidencias del equipo	Incidencia	Integer (5)	NO	NO	NULL	Causas de ingreso del equipo
Año	año	Integer (4)	NO	NO	NULL	Año en que ingreso el equipo
Orden de servicio	orden_servicio	Integer (5)	SI	NO	NOT NULL	Número de orden de servicio
Nombre del personal	usuario	Varchar (10)	NO	SI	NOT NULL	Nombre del personal que administra la BD
Fecha de incidencia del equipo	fecha	Datetime (8)	NO	NO	NULL	Fecha de incidencia del equipo celular

Tablas con las que se relaciona tab_laboratorio_incidencias

Tablas: tab_laboratorio, tab_usuarios

Tabla IV.2.4.12 Tabla Laboratorio Incidencias

Tabla Códigos de Transacción (tab_transaction_codes)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Código de transacción	cod_trans	Varchar (3)	SI	NO	NOT NULL	Código de transacción del equipo celular
Descripción de transacción	descripcion	Varchar (50)	NO	NO	NULL	Descripción de la transacción del equipo celular

Tablas con las que se relaciona tab_transaction_codes

Tablas: tab_laboratorio

Tabla IV.2.4.13 Tabla Códigos de Transacción

Tabla Problema Encontrado (tab_problem_found)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Código del problema	cod_prob_found	Varchar (5)	SI	NO	NOT NULL	Código problema encontrado
Descripción del problema	descripcion	Varchar (50)	NO	NO	NULL	Descripción del problema encontrado

Tablas con las que se relaciona tab_problem_found

Tablas: tab_laboratorio

Tabla IV.2.4.14 Tabla Problema Encontrado



Tabla Laboratorio (tab_laboratorio)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Año	año	Integer (4)	NO	NO	NULL	Año en que ingreso el equipo
Orden de servicio	orden_servicio	Integer (5)	SI	NO	NOT NULL	Número de orden de servicio
Identificador del empleo	idpersonal	Integer (7)	NO	NO	NOT NULL	Id. único del empleado
Código de transacción	cod_trans	Varchar (5)	NO	SI	NOT NULL	Código de transacción del equipo celular
msn de salida	msn_salida	Varchar (10)	NO	NO	NULL	msn de salida del equipo celular
imei de salida	imei_salida	Integer (15)	NO	NO	NULL	imei de salida del equipo celular
Código de estado de reparación	cod_rep-status	Varchar (4)	NO	SI	NOT NULL	Código de estado de reparación del equipo celular
Fecha de salida	sp_fecha	Datetime (8)	NO	NO	NOT NULL	Fecha de salida programada
Hora de salida	sp_hora	Datetime (5)	NO	NO	NOT NULL	Hora de salida programada
Fecha de recepción	r_fecha	Datetime (8)	NO	NO	NULL	Fecha de recepción del equipo celular
Hora de recepción	r_hora	Datetime (5)	NO	NO	NULL	Hora de recepción del equipo celular
Tiempo de reparación	tiempo_reparacion	Datetime (5)	NO	NO	NULL	Tiempo de reparación del equipo celular
Mojado	mojado	Char (1)	NO	NO	NOT NULL	Equipo celular mojado
Golpeado	golpeado	Char (1)	NO	NO	NOT NULL	Equipo celular golpeado
Intervenido	intervenido	Char (1)	NO	NO	NOT NULL	Equipo celular intervenido
esn_csn de salida	esn_csn_salida	Varchar (11)	NO	NO	NULL	Esn_csn de salida del equipo
Versión de software de salida	ver_sw_salida	Varchar (20)	NO	NO	NULL	Versión de salida del software del equipo celular
Nombre de la parte no disponible 1	partes_no_disponibles1	Varchar (20)	NO	NO	NULL	Nombre de la parte no disponible del equipo



Nombre de la parte no disponible 2	partes_no_disponibles2	Varchar (20)	NO	NO	NULL	Nombre de la parte no disponible del equipo
Código del problema	cod_prob_found	Varchar (5)	NO	SI	NOT NULL	Código del problema encontrado
Código del segundo problema encontrado	cod_2_prob_enc	Varchar (5)	NO	SI	NOT NULL	Código del segundo problema encontrado en el equipo
Código de reparación	cod_rep_codes	Varchar (5)	NO	SI	NOT NULL	Código de reparación del equipo celular
Código del segundo problema reparado	cod_2_prob_rep	Varchar (5)	NO	SI	NOT NULL	Código del segundo problema reparado en el equipo
Status	Status	Varchar (50)	NO	NO	NULL	Status del equipo celular

Tablas con las que se relaciona tab_laboratorio

Tablas: tab_laboratorio_incidencias, tab_transaction_codes, tab_repair_codes, tab_problem_found, tab_laboratorio_detalle, tab_repair_status_codes

Tabla IV.2.4.15 Tabla Laboratorio

Tabla Laboratorio Detalles (tab_laboratorio_detalle)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Año	Año	Integer (4)	NO	NO	NULL	Año en que ingreso el equipo
Orden de servicio	orden_servicio	Integer (5)	NO	SI	NOT NULL	Número de orden de servicio
Código de parte	cod_part_codes	Varchar (2)	NO	SI	NOT NULL	Código de parte del equipo celular
Cantidad reemplazada	cant_reempla	Integer (5)	NO	NO	NULL	Cantidad reemplazada en el equipo celular
Cantidad intercambiada	cant_intercam	Integer (5)	NO	NO	NULL	Cantidad intercambiada del equipo celular
Código de falla de parte	cod_falla_parte	Varchar (5)	NO	SI	NOT NULL	Código de falla de parte del celular



Análisis, Diseño y Construcción del Sistema

Código de referencia	cod_ref_codes	Varchar (5)	NO	SI	NOT NULL	Código de referencia del equipo celular
Sin código de referencia	no_desgdor_ref	Varchar (5)	NO	NO	NULL	Sin código de referencia del equipo celular
Resoldar Equipo	resoldar_reconstruccion	Varchar (20)	NO	NO	NULL	Resoldar parte del equipo celular
Observaciones	Observaciones	Varchar (50)	NO	NO	NULL	Observación considerada del personal

Tablas con las que se relaciona tab_laboratorio_detalle

Tablas: tab_laboratorio, tab_parts_codes, tab_parts_failure_codes, tab_references_codes

Tabla IV.2.4.16 Tabla Laboratorio Detalles

Tabla Códigos de Reparación (tab_repair_codes)

Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Código de reparación	cod_rep_codes	Varchar (5)	SI	NO	NOT NULL	Código de reparación del equipo celular
Descripción de la reparación	descripcion	Varchar (50)	NO	NO	NULL	Descripción de reparación del equipo celular

Tablas con las que se relaciona tab_repair_codes

Tablas: tab_laboratorio

Tabla IV.2.4.17 Tabla Códigos de Reparación

Tabla Reparar Estatus Código (tab_repair_status_codes)

Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Código de estado de reparación	cod_rep_status	Varchar (4)	SI	NO	NOT NULL	Código de estado de reparación del equipo celular
Descripción del código del estado de reparación	descripcion	Varchar (50)	NO	NO	NULL	Descripción de reparación del equipo
Descripción del estado reparación	description	Varchar (50)	NO	NO	NULL	Descripción del estado de reparación



Nivel de reparación	Nivel	Numeric (1)	NO	NO	NULL	Nivel de reparación del equipo
---------------------	-------	-------------	----	----	------	--------------------------------

Tablas con las que se relaciona tab_repair_status_codes

Tablas: tab_laboratorio

Tabla IV.2.4.18 Tabla Reparar Estatus Código

Tabla Código de Partes (tab_parts_codes)

Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Código de parte	cod_part_codes	Varchar (2)	SI	NO	NOT NULL	Código de parte del equipo celular
Descripción de la parte	descripcion	Varchar (50)	NO	NO	NULL	Descripción de parte del equipo celular

Tablas con las que se relaciona tab_parts_codes

Tablas: tab_laboratorio_detalle

Tabla IV.2.4.19 Tabla Código de Partes

Tabla Código de Partes Dañadas (tab_part_failure_codes)

Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Código de falla	cod_falla_parte	Varchar (5)	SI	NO	NOT NULL	Código de falla de la parte dañada
Descripción de la falla	descripcion	Varchar (50)	NO	NO	NULL	Descripción de la falla de la parte dañada

Tablas con las que se relaciona tab_part_failure_codes

Tablas: tab_laboratorio_detalle

Tabla IV.2.4.20 Tabla Código de Partes Dañadas

Tabla Código de Referencia (tab_references_codes)

Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Código de referencia	cod_ref_codes	Varchar (5)	SI	NO	NOT NULL	Código de referencia del equipo celular
Descripción	descripcion	Varchar (50)	NO	NO	NULL	Descripción de referencia del equipo

Tablas con las que se relaciona tab_references_codes

Tablas: tab_laboratorio_detalle

Tabla IV.2.4.21 Tabla Código de Referencia



Tabla Entrega (tab_entrega)						
Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Año	año	Integer (4)	NO	NO	NULL	Año en que ingreso el equipo
Orden de servicio	orden_servicio	Integer (5)	SI	NO	NOT NULL	Número de orden de servicio
Fecha de salida	fecha_salida	Datetime (8)	NO	NO	NOT NULL	Fecha de salida del equipo celular
Hora de salida	hora_salida	Datetime (5)	NO	NO	NOT NULL	Hora de salida del equipo celular
Tiempo total	tiempo_total	Datetime (5)	NO	NO	NULL	Tiempo total de reparación equipo celular
Numero de recibo	no_recibo	Integer (5)	NO	NO	NULL	Número de recibo
Entregado por	entrega_por	Varchar (50)	NO	NO	NULL	Nombre de la persona que entrega el equipo celular
Recibido por	persona_recibe	Varchar (50)	NO	NO	NULL	Nombre de la persona que recibe el equipo celular
Numero de Factura	no_factura	Integer (20)	NO	NO	NULL	Número de la factura del servicio
Estado de reparación	edo_reparacion	Varchar (50)	NO	NO	NOT NULL	Estado de reparación del equipo celular
Mensajería	Mensajería	Varchar (50)	NO	NO	NULL	Envío por mensajería del equipo celular
Numero de guía de salida	no_guia_salida	Integer (20)	NO	NO	NULL	Número de guía de salida del equipo
Tarjeta de crédito	tarjeta_credito	Varchar (30)	NO	NO	NULL	Número de la tarjeta de crédito del cliente
Tipo de la tarjeta de crédito	tipo_tarjeta_credito	Varchar (20)	NO	NO	NULL	Tipo de la tarjeta de crédito del cliente
Fecha de expiración	fecha_expiracion	Datetime (5)	NO	NO	NULL	Fecha de expiración de la tarjeta de crédito



Análisis, Diseño y Construcción del Sistema

Autorizado	autoriza	Varchar (50)	NO	NO	NULL	Persona que autoriza el uso de la tarjeta de crédito
Observaciones	observaciones	Varchar (50)	NO	NO	NULL	Observaciones que considere el personal
Costo	valor	Numeric (10)	NO	NO	NULL	Costo de la reparación del equipo celular
Estatus final	status_final	Varchar (50)	NO	NO	NULL	Estatus final del equipo celular

Tablas con las que se relaciona tab_entrega

Tablas: tab_entrega, incidencias

Tabla IV.2.4.22 Tabla Entrega

Tabla Entrega Incidencias (tab_entrega_incidencias)

Nombre	Acrónimo	Tipo de Dato	PK	FK	Null / Not Null	Descripción
Incidencias del equipo	incidencia	Integer (5)	NO	NO	NULL	Causas de ingreso del equipo
Orden de servicio	orden_servicio	Integer (5)	NO	SI	NOT NULL	Número de orden de servicio
Nombre del personal	usuario	Varchar (10)	NO	SI	NOT NULL	Nombre del personal que administra la BD
Fecha de incidencia del equipo	fecha	Datetime (8)	NO	NO	NULL	Contiene la fecha de incidencia del equipo

Tablas con las que se relaciona tab_entrega_incidencias

Tablas: tab_entrega, tab_usuarios

Tabla IV.2.4.23 Tabla Entrega Incidencias

IV.2.5 Diagrama Entidad Relación

El Diagrama Entidad Relación (DER) es un modelo que describe con un alto nivel de abstracción la distribución de datos almacenados en un sistema.

Representa a la base de datos a través de un esquema gráfico empleando la terminología de entidades, los cuales son objetos que existen y son los elementos principales que se identifican en el problema a resolver con el diagramado y se distinguen de otros por sus características particulares denominadas atributos, el enlace que rige la unión de las entidades esta representada por la relación del modelo.



Para establecer una relación entre dos entidades se cuenta con una llave primaria, que es un campo dentro de la entidad correspondiente que servirá como llave de acceso a la tupla, por lo que no deberá ser nulo y que existe en otra entidad como llave foránea estableciendo de esta forma, la asociación entre ellas.

El Diagrama Entidad Relación es multidireccional, esto es, puede leerse siguiendo cualquier dirección. Y no muestra cardinalidad, es decir, no muestra el número de objetos que participan en la relación. En la figura IV.2.5.1 se muestra el Diagrama Entidad Relación del sistema.

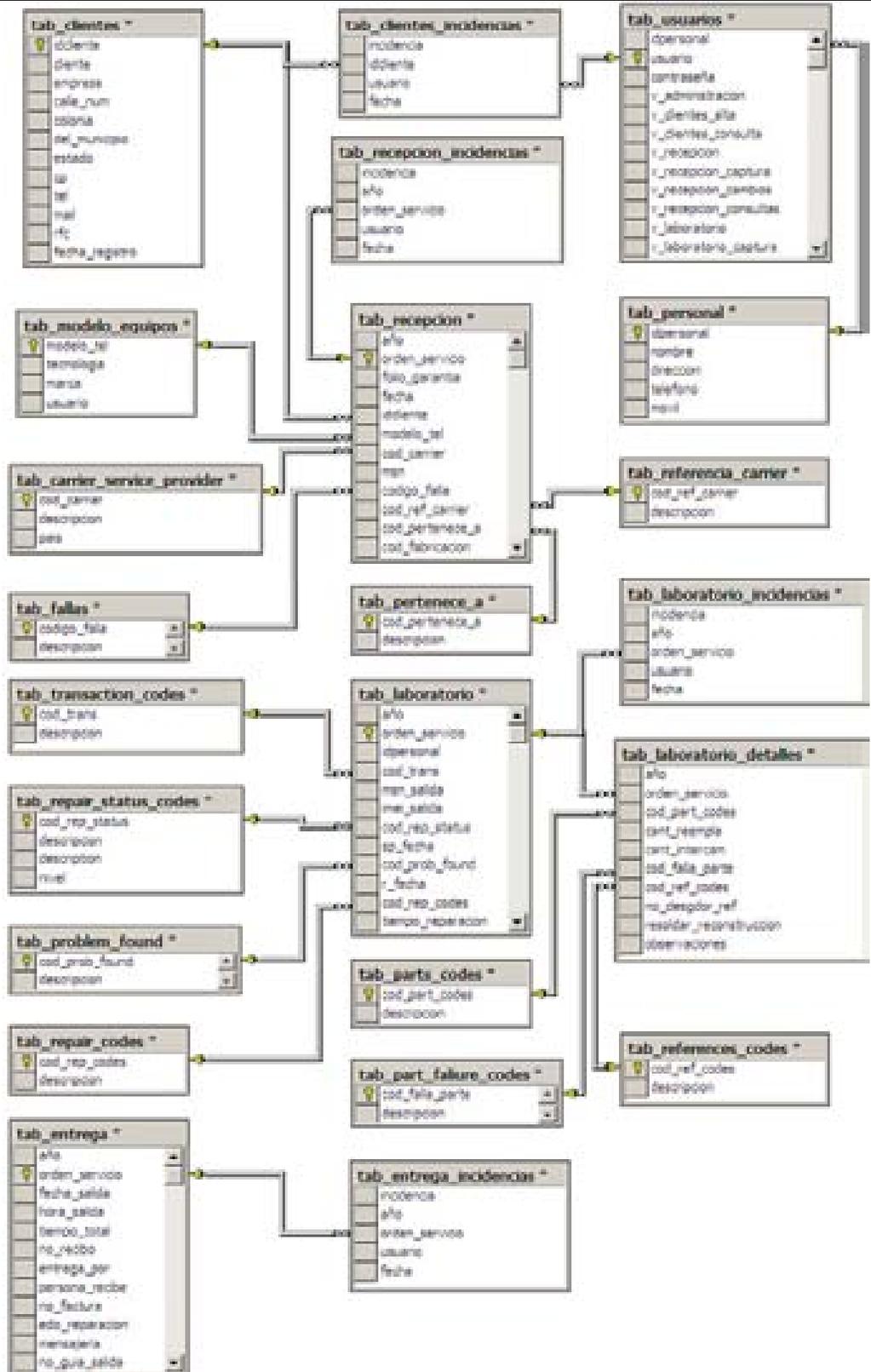


Fig. IV.2.5.1 Diagrama Entidad Relación



IV.2.6 Normalización

La normalización es el proceso mediante el cual se transforman datos complejos a un conjunto de estructuras de datos más pequeñas, que además de ser más simples y estables, son fáciles de mantener. También contribuye a minimizar los problemas de lógica. Cada regla está basada en la que le antecede.

Las ventajas que se obtienen tras la normalización de datos son:

- Facilidad de uso. Los datos están agrupados en tablas que identifican claramente un objeto o una relación.
- Flexibilidad. La información que necesitan los usuarios se puede obtener de las tablas relacionales o relaciones mediante las operaciones de álgebra relacional.
- Precisión. Las interrelaciones entre tablas consiguen mantener información diferente relacionada con toda exactitud.
- Seguridad. Los controles de acceso para consultar o actualizar información son mucho más sencillos de implementar.
- Facilidad de implementación. Las tablas se almacenan físicamente como archivos planos.
- Independencia de datos. Los programas no están ligados a las estructuras, con lo que se consigue aumentar la base de datos añadiendo nuevos atributos o nuevas tablas sin que afecten a los programas que las usan.
- Claridad. La representación de la información es clara y sencilla para el usuario, son tablas simples.
- Facilidad de gestión. Los lenguajes manipulan la información de forma sencilla al estar los datos basados en el álgebra y cálculo relacional.
- Mínima redundancia. La información no estará duplicada innecesariamente dentro de las estructuras.



- Máximo rendimiento de las aplicaciones. Sólo se trata aquella información que va a ser de utilidad a cada aplicación.

Formas Normales

Existen varios niveles de normalización, sin embargo para este proyecto se utilizó la primera forma normal (1NF), la segunda forma normal (2FN) y la tercera forma normal (3FN), porque estas tres proveen suficiente nivel de normalización para cumplir con las necesidades de la base de datos del sistema.

Recordemos que para que una tabla este en un nuevo nivel o forma de normalización, es necesario que haya cumplido con la regla anterior.

En la tabla IV.2.6.1 se describe brevemente en que consiste cada una de las reglas y posteriormente se explicarán con más detalle utilizando una tabla del proyecto.

Forma Normal	Descripción
Primera (1FN)	Incluye la eliminación de todos los grupos repetidos.
Segunda (2FN)	Asegura que todas las columnas que no son llave sean completamente dependientes de la llave primaria.
Tercera (3FN)	Elimina cualquier dependencia transitiva. Una dependencia transitiva es aquella en la cual las columnas que no son llave son dependientes de otras columnas que tampoco son llave

Tabla IV.2.6.1 Niveles de normalización

A la tabla IV.2.6.2 que no esta normalizada le aplicaremos las reglas de normalización.



orden_servicio	incidencia	idcliente	cliente	RFC	modelo_tel	tecnología	marca	cod_falla	descripcion_falla
2008-1	ALTA	3567	Virginia	LOGV812103	T191	GSM	MOTOROLA	F	FAILURE
2008-1	ALTA	3567	Virginia	LOGV812103	T191	GSM	MOTOROLA	F	FAILURE
2008-1	ALTA	3567	Virginia	LOGV812103	T191	GSM	MOTOROLA	F	FAILURE
2008-2	MODIFICADA	3689	Arturo	MEMA751108	V60T COLOR	TDMA	NOKIA	P	PREVENTATIVE MAINTENANCE
2008-2	MODIFICADA	3689	Arturo	MEMA751108	V60T COLOR	TDMA	NOKIA	P	PREVENTATIVE MAINTENANCE
2008-3	BAJA	3793	Leandro	MIEL801107	V810	CDMA	SAMSUNG	CS	CUSTOMER SATISFACTION ISSUE
2008-3	BAJA	3793	Leandro	MIEL801107	V810	CDMA	SAMSUNG	CS	CUSTOMER SATISFACTION ISSUE

Tabla IV.2.6.2 Tabla sin normalizar

Primera Forma Normal (1FN)

Establece que las columnas repetidas deben eliminarse y colocarse en tablas separadas.

Los pasos a seguir son:

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

Los registros quedan ahora conformados en dos tablas que llamaremos Clientes (ver tabla IV.2.6.3) y Recepción (ver tabla IV.2.6.4)

orden_servicio	incidencia	idcliente	cliente	RFC
2008-1	alta	3567	Virginia	LOGV812103
2008-2	modificada	3689	Arturo	MEMA751108
2008-3	baja	3793	Leandro	MIEL801107

Tabla IV.2.6.3 Clientes



orden_servicio	modelo_tel	tecnología	marca	cod_falla	descripcion_falla
2008-1	T191	GSM	MOTOROLA	F	FAILURE
2008-1	T191	GSM	MOTOROLA	F	FAILURE
2008-1	T191	GSM	MOTOROLA	F	FAILURE
2008-2	V60T COLOR	TDMA	NOKIA	P	PREVENTATIVE MAINTENANCE
2008-2	V60T COLOR	TDMA	NOKIA	P	PREVENTATIVE MAINTENANCE
2008-3	V810	CDMA	SAMSUNG	CS	CUSTOMER SATISFACTION ISSUE
2008-3	V810	CDMA	SAMSUNG	CS	CUSTOMER SATISFACTION ISSUE

Tabla IV.2.6.4 Recepción

Segunda Forma Normal (2FN)

En esta regla todas las dependencias parciales deben eliminarse y separarse dentro de sus propias tablas. Una dependencia parcial es un término que describe aquellos datos que no dependen de la llave primaria para identificarlos.

Los pasos a seguir son:

- Determinar cuales columnas que no son llave no dependen de la llave primaria.
- Eliminar esas columnas de la tabla.
- Crear una segunda tabla con esas columnas y la columna de la llave primaria de la cual dependen.



La tabla Clientes (ver tabla IV.2.6.3) esta en 2FN, porque cualquier valor único de orden_servicio determina un solo valor para cada columna, es decir todas las columnas son dependientes de la llave primaria orden_servicio.

La tabla Recepción no se encuentra en 2FN, ya que las columnas tecnología y marca son dependientes de modelo_tel. Y descripción_falla es dependiente de cod_falla, pero no son dependientes de orden_servicio. Por lo tanto, eliminaremos esas columnas de la tabla Recepción (ver tabla IV.2.6.5) y crearemos las tablas llamadas Modelo_Equipo (ver tabla IV.2.6.6) y Falla_Equipo (ver tabla IV.2.6.7) con dichas columnas y la llave primaria de la cual dependen.

orden_servicio	modelo_tel	cod_falla
2008-1	T191	F
2008-1	T191	F
2008-1	T191	F
2008-2	V60T COLOR	P
2008-2	V60T COLOR	P
2008-3	V810	CS
2008-3	V810	CS

Tabla IV.2.6.5 Recepción

modelo_tel	tecnologia	marca
T191	GSM	MOTOROLA
V60T COLOR	TDMA	NOKIA
V810	CDMA	SAMSUNG
FX 1500	ANALOGICO	LG

Tabla IV.2.6.6 Modelo_Equipo

cod_falla	descripcion_falla
A	ABUSE
CS	CUSTOMER SATISFACTION ISSUE
F	FAILURE
P	PREVENTATIVE MAINTENANCE

Tabla IV.2.6.7 Falla_Equipo



Tercera Forma Normal (3FN)

Una tabla esta normalizada en esta forma si todas las columnas que no son llave son funcionalmente dependientes por completo de la llave primaria y no hay dependencias transitivas. Una dependencia transitiva es aquella en la cual existen columnas que no son llaves que dependen de otras columnas que tampoco son llaves.

Los pasos a seguir son:

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

Las tablas recepción (ver tabla IV.2.6.5), modelo_equipo (ver tabla IV.2.6.6) y falla_equipo (ver tabla IV.2.6.7) se encuentran en 3FN, sin embargo la tabla Clientes (ver tabla IV.2.6.3) no lo esta, ya que cliente y rfc son dependientes de idcliente y esta columna no es la llave primaria.

Para normalizar la tabla Clientes (ver tabla IV.2.6.8), moveremos las columnas no llave y la columna llave de la cual dependen dentro de una nueva tabla llamada Recepción Incidencias (ver tabla IV.2.6.9).

idcliente	cliente	RFC
3567	Virginia	LOGV812103
3689	Arturo	MEMA751108
3793	Leandro	MIEL801107

Tabla IV.2.6.8 Clientes



orden_ servicio	incidencia
2008-1	ALTA
2008-2	MODIFICADA
2008-3	BAJA

Tabla IV.2.6.9 Recepción Incidencias

IV.2.7 Construcción física de la base de datos

Para la construcción de la base de datos se consideraron los requerimientos reflejados en el diagrama entidad-relación, con el cual se definió el esquema de la base de datos en un archivo de texto, el cual contiene todas las sentencias SQL que se requieren para definir las tablas, sus atributos, los tipos de datos, los índices, los campos, etc.

La razón para hacerlo de esta manera es que durante el desarrollo de la aplicación es muy frecuente borrar la base de datos y crearla de nuevo, por lo que teniendo el archivo de texto nos facilita mucho las cosas.

Construcción de la base de datos

Se selecciono SQL Server 2000 como el manejador de nuestro sistema por que es un sofisticado manejador de bases de datos relacionales, además que puede soportar todas las instrucciones SQL, incluyendo sub-consultas, transacciones, funciones y tipos de datos definidos por el usuario.



SQL Server 2000 incluye dos herramientas que nos permiten trabajar de una manera más sencilla con la base de datos, el Enterprise Manager y el Query Analyser.

Primeramente utilizaremos la herramienta Enterprise Manager, la cual al iniciar la aplicación nos presenta una ventana donde se indica el nombre del servidor y las carpetas que contienen las bases de datos entre otros.

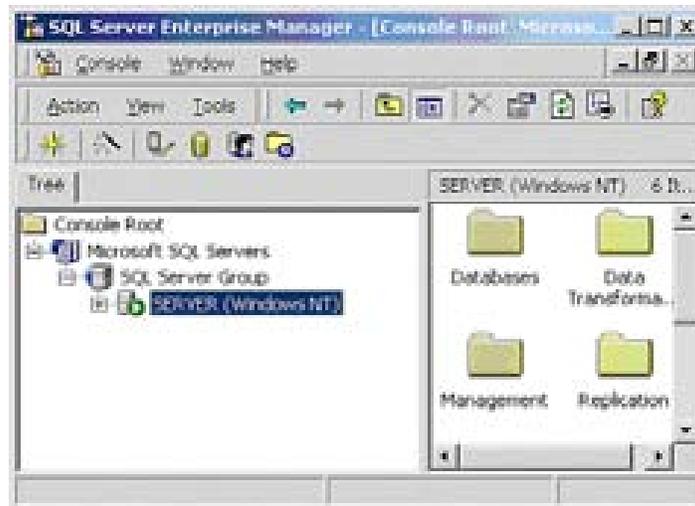


Figura IV.2.7.1 Enterprise Manager de SQL

Cabe destacar que al instalar SQL Server 2000, se instala una base de datos maestra, dentro de la que se encuentra la tabla denominada sysdatabases, la cual contiene una lista de las bases de datos que contiene el servidor SQL.

Para poder crear nuestra base de datos haremos click con el boton derecho del mouse y seleccionaremos nueva base de datos, acto seguido nos aparecerá un cuadro con las propiedades que tendrá nuestra base de datos.

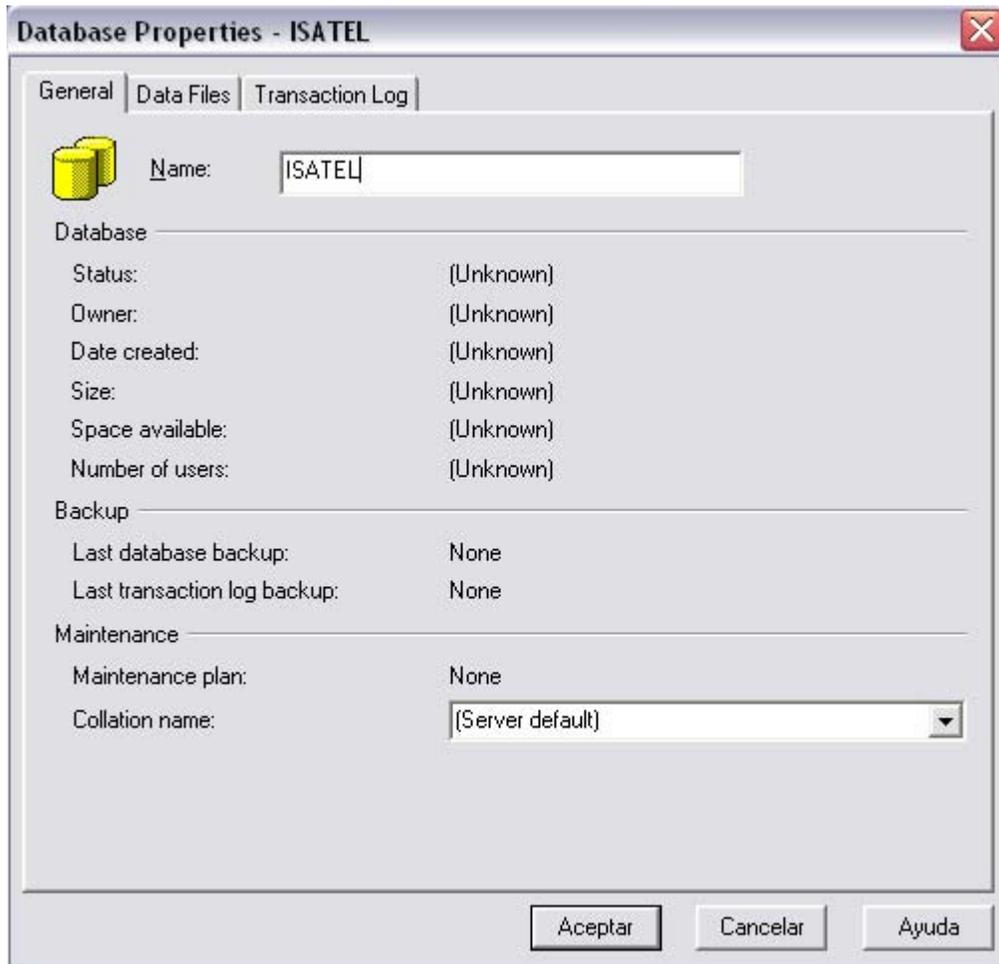


Figura IV.2.7.1 Cuadro de propiedades de la base de datos

Para poder crear nuestra base de datos haremos click con el botón derecho del mouse y seleccionaremos nueva base de datos, acto seguido nos aparecerá un cuadro con las propiedades que tendrá nuestra base de datos.

Como se puede observar dentro de este cuadro se le asignara el nombre a nuestra base de datos, la ruta en donde se guardara (por default en C:\Archivos de Programa\MicrosoftSQLServer\MSSQL\data\MyDatabase_Data.MDF).



La base de datos ha sido creada, ahora procederemos a crear las tablas que contendrá. Para lograr esto haremos click con el botón derecho del mouse en tablas y seleccionaremos: nueva opción de tabla, esto iniciará el diseñador de tablas que nos permite añadir, editar y eliminar campos de la tabla o tablas a crear.

Cada fila en el diseñador de tablas representa una columna de la tabla a crear, para ejemplificar crearemos una tabla que contiene los detalles de los clientes que utilizan nuestro sistema.

El nombre de la columna, es el nombre por el cual haremos referencia a cada columna particular en la tabla. Es importante notar que al seleccionar el tipo de dato, los campos de longitud y permitir valores nulos son completados automáticamente. Para asegurarnos que cada usuario de la tabla es único haremos del campo ID la llave primaria.

Nombre de columna	Tipo de datos	Longitud	Permitir valores nulos
cliente	varchar	60	✓
empresa	varchar	60	✓
calle_num	varchar	50	✓
colonia	varchar	50	✓
del_municipio	varchar	50	✓
estado	varchar	30	✓
cp	varchar	6	✓
tel	varchar	20	✓
mail	varchar	20	✓
rfc	varchar	15	✓
fecha_registro	varchar	8	✓

Figura IV.2.7.3 Diseñador de Tablas – Tabla de Clientes



Creación de las tablas utilizando Query Analyzer

Para la creación de las tablas restantes de nuestra base de datos utilizaremos la herramienta Query Analyzer por ser esta mucho más rápida de utilizar, dado que se requiere introducir las sentencias SQL requeridas en la ventana del Query Analyzer para crear las tablas.

En seguida se muestran las sentencias SQL necesarias para la creación de la tabla denominada *tab_laboratorio*:

```
CREATE TABLE [dbo].[tab_laboratorio] (  
    [año] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [orden_servicio] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [idpersonal] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [cod_reparacion] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [msn_salida] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [imei_salida] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [edo_reparacion] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [sp_fecha] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [sp_hora] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [r_fecha] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [r_hora] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [tiempo_reparacion] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [mojado] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
    [golpeado] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
```



```
[intervenido] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[esn_csn_salida] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[ver_sw_salida] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[partes_no_disponibles1] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[partes_no_disponibles2] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[cod_1_prob_enc] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[cod_2_prob_enc] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[cod_1_prob_rep] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[cod_2_prob_rep] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[status] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL  
  
) ON [PRIMARY]  
  
GO
```

Es importante mencionar que para poder ejecutar las sentencias SQL es necesario presionar la tecla F5, o bien hacer clic en el botón verde. Lo cual nos mostrará el resultado siguiente en la parte inferior de la ventana del Query Analyzer, como se muestra en la figura IV.2.7.4.

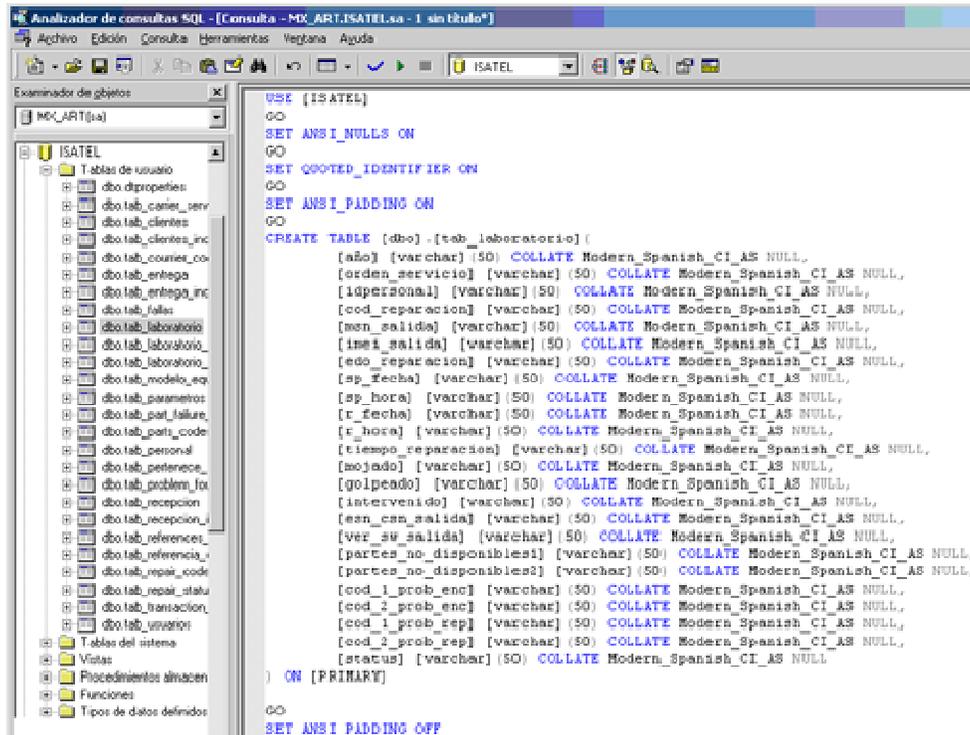


Figura IV.2.7.4 Analizador de Consultas – Tabla Laboratorio

En la figura IV.2.7.5 podemos apreciar que la tabla tab_laboratorio ha sido creada, también podemos ver los detalles de la misma, los campos que la componen, el nombre de la columna y el tipo de datos.



Nombre de columna	Tipo de datos	Permitir n...
año	varchar(50)	<input checked="" type="checkbox"/>
orden_servicio	varchar(50)	<input checked="" type="checkbox"/>
idpersonal	varchar(50)	<input checked="" type="checkbox"/>
cod_reparacion	varchar(50)	<input checked="" type="checkbox"/>
mon_salida	varchar(50)	<input checked="" type="checkbox"/>
mes_salida	varchar(50)	<input checked="" type="checkbox"/>
edo_reparacion	varchar(50)	<input checked="" type="checkbox"/>
sp_fecha	varchar(50)	<input checked="" type="checkbox"/>
sp_hora	varchar(50)	<input checked="" type="checkbox"/>
r_fecha	varchar(50)	<input checked="" type="checkbox"/>
r_hora	varchar(50)	<input checked="" type="checkbox"/>
tiempo_reparacion	varchar(50)	<input checked="" type="checkbox"/>
mojado	varchar(50)	<input checked="" type="checkbox"/>
golpeado	varchar(50)	<input checked="" type="checkbox"/>
intervenido	varchar(50)	<input checked="" type="checkbox"/>
esn_con_salida	varchar(50)	<input checked="" type="checkbox"/>
ver_con_salida	varchar(50)	<input checked="" type="checkbox"/>
partes_no_disponibles1	varchar(50)	<input checked="" type="checkbox"/>
partes_no_disponibles2	varchar(50)	<input checked="" type="checkbox"/>
cod_1_prob_enc	varchar(50)	<input checked="" type="checkbox"/>
cod_2_prob_enc	varchar(50)	<input checked="" type="checkbox"/>
cod_1_prob_rep	varchar(50)	<input checked="" type="checkbox"/>
cod_2_prob_rep	varchar(50)	<input checked="" type="checkbox"/>
status	varchar(50)	<input checked="" type="checkbox"/>

Figura IV.2.7.5 Detalles de la Tabla Laboratorio

Continuaremos aplicando el mismo procedimiento para crear cada una de las tablas que componen nuestra base de datos, escribiremos el código SQL necesario dentro del Query Analyzer para la creación de cada una de las tablas.

```
CREATE TABLE [dbo].[tab_clientes] (
```

```
    [id] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
```

```
    [cliente] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
```

```
    [empresa] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
```

```
    [calle_num] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
```

```
    [colonia] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
```

```
    [del_municipio] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL,
```



```
[estado] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
  
[cp] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
  
[tel] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
  
[mail] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
  
[rfc] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
  
[fecha_registro] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL  
  
) ON [PRIMARY]  
  
GO  
  
CREATE TABLE [dbo].[tab_personal] (  
  
    [idpersonal] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
  
    [nombre] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
  
    [direccion] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
  
    [telefono] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
  
    [movil] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL  
  
) ON [PRIMARY]  
  
GO  
  
CREATE TABLE [dbo].[tab_recepcion] (  
  
    [año] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
  
    [orden_servicio] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
  
    [folio_garantia] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
```



[fecha] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[hora] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[modelo_tel] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[imei] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[msn] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[esn_csn] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[cod_tranceptor] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[cod_apc] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[cod_fabricacion] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[id_origen] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[cod_carrier] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[ver_sw_entrada] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[no_min] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[no_min_acum] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[tarjeta_sim] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[no_guia_entrada] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[goma] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[cargador] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[bateria] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[antena] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,



[tapa] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[hotler] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[no_cim] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[garantia_factura] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[fecha_factura] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[cod_apc_accesorio] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[codigo_falla] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[garantia] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[costo] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[valor] [char] (10) COLLATE Modern_Spanish_CI_AS NULL ,
[no_reincidencias] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[observaciones] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[recibido_por] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[id_cliente] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[cliente] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[empresa] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[calle_num] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[colonia] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[del_municipio] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,
[estado] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,



```
[cp] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[tel] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[mail] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[status] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[memoria] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[memoria_capacidad] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[referencia_carrier] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[tapon] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[pertenece_a] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[cotizar] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[manos_libres] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[interfaz] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL ,  
[bluetooth] [varchar] (50) COLLATE Modern_Spanish_CI_AS NULL  
)  
ON [PRIMARY]  
GO
```

Cuando todas las tablas de nuestra base de datos hallan sido creadas podremos observar cada una de ellas.

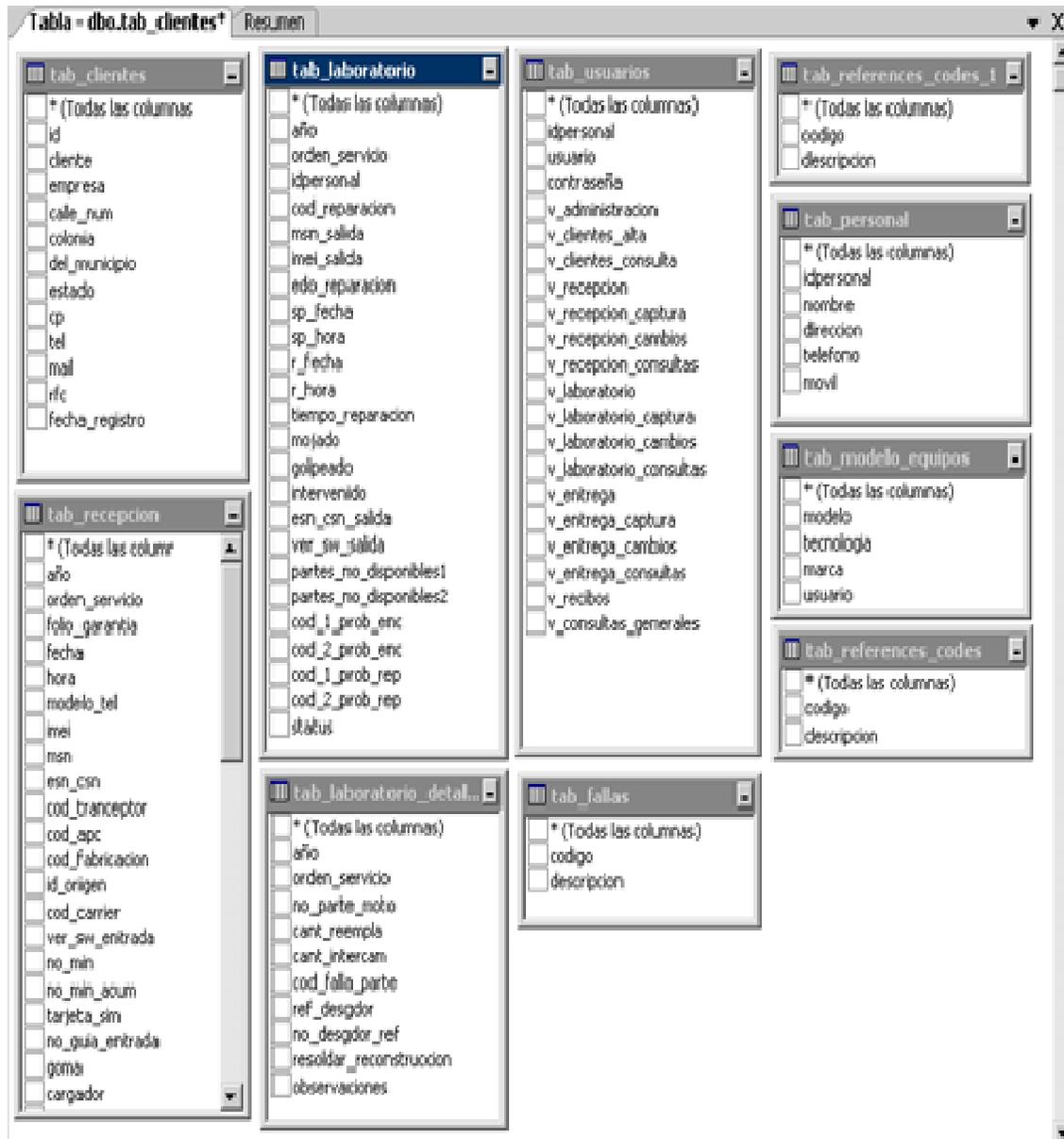


Figura IV.2.7.6 Tablas de la Base de Datos



Consultas a la base de datos

Una consulta es el método para acceder a los datos en nuestra base de datos. Con las consultas se puede modificar, borrar, mostrar y agregar datos en la misma. Para esto se utiliza un lenguaje de consultas SQL.

Técnicamente hablando, las consultas a la base de datos se realizan a través de un lenguaje de manipulación de datos (DML – Data Manipulation Language). SQL es un lenguaje DML, pero además posee otras características de otros lenguajes.

Utilizamos las sentencias SQL Select from , update en diversas de nuestras tablas para hacer una consulta a una tabla, y a su vez actualizar los datos, como se puede observar en el código sql de la tabla cambio_contraseña, en este caso en particular podemos observar que se selecciona de la tab_usuarios todos los campos de la misma.

```
SQLAplicacion = "SELECT * FROM tab_usuarios WHERE usuario LIKE '" &  
usuario & "' and contraseña LIKE '" & contraseña_actual & "' AND idpersonal LIKE  
'" & idpersonal_usuario & """
```

```
Set rsdBusca = New ADODB.Recordset  
rsdBusca.CursorLocation = adUseClient  
rsdBusca.CursorType = adOpenKeyset  
rsdBusca.LockType = adLockOptimistic  
rsdBusca.Source = sSQLAplicacion  
rsdBusca.ActiveConnection = cnd  
rsdBusca.Open  
regencontrados = rsdBusca.RecordCount
```

```
If regencontrados = 0 Then  
    MsgBox "La contraseña actual es incorrecta", vbExclamation  
    Text1.Text = ""  
    Text1.SetFocus  
Else  
    If contraseña_nueva <> confirmar_contraseña Then  
        MsgBox "La nueva contraseña no coincide con la confirmación",  
        vbExclamation  
        Text2.Text = ""  
        Text3.Text = ""
```



```
Text2.SetFocus
Else
'ACTUALIZA CONTRASEÑA
Dim orden As String
Set actualiza = New Command
orden = "UPDATE tab_usuarios SET contraseña = " &
confirmar_contraseña & " WHERE usuario LIKE " & usuario & " AND idpersonal
LIKE " & idpersonal_usuario & ""
With actualiza
.ActiveConnection = cnd
.CommandText = orden
.CommandType = adCmdText
End With
actualiza.Execute nregs

MsgBox "Contraseña actualizada."
Unload Me
End If
```

Para la actualización de los datos SQL dispone de la sentencia **UPDATE**. La sentencia **UPDATE** permite la actualización de uno o varios registros de una única tabla.

Otro ejemplo lo tenemos al momento de la captura de los datos del cliente en la tabla Clientes_Recepcion_Captura, en la que se realizan diversas consultas relacionadas con los datos del mismo (cliente, empresa, domicilio, entidad federativa, R.F.C, mail, etc.), lo anterior lo podemos observar en el siguiente código.

```
Private Sub busca_datos_cliente()
cliente = List1.Text
empresa = List1.Text
Sql = "SELECT * FROM tab_clientes WHERE cliente LIKE " & cliente & ""
Set busca = New ADODB.Recordset
busca.CursorLocation = adUseClient
busca.CursorType = adOpenStatic
busca.LockType = adLockOptimistic
busca.Source = Sql
busca.ActiveConnection = cnd
busca.Open
```



```
regencontrados = busca.RecordCount

If regencontrados = 0 Then
  Sql8 = "SELECT * FROM tab_clientes WHERE empresa LIKE '" & empresa
  & """"
  Set busca8 = New ADODB.Recordset
  busca8.CursorLocation = adUseClient
  busca8.CursorType = adOpenStatic
  busca8.LockType = adLockOptimistic
  busca8.Source = Sql8
  busca8.ActiveConnection = cnd
  busca8.Open
  regencontrados8 = busca8.RecordCount

  If regencontrados8 = 0 Then
    Else
      Recepcion.Label43.Caption = busca8("id")
      Recepcion.Text26.Text = busca8("cliente")
      Recepcion.Text27.Text = busca8("empresa")
      Recepcion.Text28.Text = busca8("calle_num")
      Recepcion.Text29.Text = busca8("colonia")
      Recepcion.Text30.Text = busca8("del_municipio")
      Recepcion.Text31.Text = busca8("estado")
      Recepcion.Text32.Text = busca8("cp")
      Recepcion.Text33.Text = busca8("tel")
      Recepcion.Text34.Text = busca8("mail")
      Recepcion.Text16.Text = busca8("rfc")
    End If
  busca8.Close
Else
  Recepcion.Label43.Caption = busca("id")
  Recepcion.Text26.Text = busca("cliente")
  Recepcion.Text27.Text = busca("empresa")
  Recepcion.Text28.Text = busca("calle_num")
  Recepcion.Text29.Text = busca("colonia")
  Recepcion.Text30.Text = busca("del_municipio")
  Recepcion.Text31.Text = busca("estado")
  Recepcion.Text32.Text = busca("cp")
  Recepcion.Text33.Text = busca("tel")
  Recepcion.Text34.Text = busca("mail")
  Recepcion.Text16.Text = busca("rfc")
End If
busca.Close
End Sub
```



```
Private Sub Option1_Click()
```

```
List1.Clear  
Text1.Text = ""  
Text2.Text = ""  
Text1.Enabled = True  
Text2.Enabled = False  
Text1.BackColor = &HFFFFFFF  
Text2.BackColor = &HE0E0E0  
Text1.SetFocus
```

```
End Sub
```

```
Private Sub Option2_Click()
```

```
List1.Clear  
Text1.Text = ""  
Text2.Text = ""  
Text1.Enabled = False  
Text2.Enabled = True  
Text1.BackColor = &HE0E0E0  
Text2.BackColor = &HFFFFFFF  
Text2.SetFocus
```

```
End Sub
```

```
Private Sub Text1_KeyUp(KeyCode As Integer, Shift As Integer)
```

```
tecla = KeyCode  
If tecla = 27 Then  
Unload Me  
Recepcion.Enabled = True  
Else  
cadena = UCase(Text1.Text)  
List1.Clear  
cliente = cadena & "%"
```

```
Sql = "SELECT * FROM tab_clientes WHERE cliente LIKE '" & cliente & """  
Set busca = New ADODB.Recordset  
busca.CursorLocation = adUseClient  
busca.CursorType = adOpenKeyset  
busca.LockType = adLockOptimistic  
busca.Source = Sql  
busca.ActiveConnection = cnd  
busca.Open  
regencontrados = busca.RecordCount
```

```
If regencontrados = 0 Then  
Else  
For contador = 0 To regencontrados - 1
```



```
cliente = UCase((busca("cliente")))
List1.AddItem cliente

    busca.MoveNext
    Next contador
End If
busca.Close
End If
End Sub

Private Sub Text2_KeyUp(KeyCode As Integer, Shift As Integer)
    tecla = KeyCode
    If tecla = 27 Then
        Unload Me
        Recepcion.Enabled = True
    Else
        cadena = UCase(Text2.Text)
        List1.Clear
        empresa = cadena & "%"

        Sql = "SELECT * FROM tab_clientes WHERE empresa LIKE '" & empresa &
,,,,
        Set busca = New ADODB.Recordset
        busca.CursorLocation = adUseClient
        busca.CursorType = adOpenKeyset
        busca.LockType = adLockOptimistic
        busca.Source = Sql
        busca.ActiveConnection = cnd
        busca.Open
        regencontrados = busca.RecordCount

        If regencontrados = 0 Then
        Else
            For contador = 0 To regencontrados - 1
                empresa = UCase((busca("empresa")))
                List1.AddItem empresa

                busca.MoveNext
            Next contador
        End If
        busca.Close
    End If
End Sub
```



En otra sección del código podemos observar como se declaran diversos **procedimientos** para el formulario de recepción de teléfonos celulares, el cual nos permite la captura de los datos del teléfono celular, entre estos el modelo, el proveedor del servicio, el código que identifica al mismo y finalmente el tipo de falla.

```
Private Sub carga_combo_modelo_equipos()  
    Combo1.Clear  
    Sql = "SELECT * FROM tab_modelo_equipos ORDER BY modelo"  
    Set busca = New ADODB.Recordset  
    busca.CursorLocation = adUseClient  
    busca.CursorType = adOpenKeyset  
    busca.LockType = adLockOptimistic  
    busca.Source = Sql  
    busca.ActiveConnection = cnd  
    busca.Open  
    regencontrados = busca.RecordCount  
  
    If regencontrados = 0 Then  
    Else  
        For contador = 0 To regencontrados - 1  
            modelo = busca("modelo")  
            Combo1.AddItem modelo  
            busca.MoveNext  
        Next contador  
    End If  
    busca.Close  
End Sub  
  
Private Sub carga_combo_carrier_service_provider()  
    Combo3.Clear  
    Sql = "SELECT * FROM tab_carrier_service_provider ORDER BY descripcion"  
    Set busca = New ADODB.Recordset  
    busca.CursorLocation = adUseClient  
    busca.CursorType = adOpenKeyset  
    busca.LockType = adLockOptimistic  
    busca.Source = Sql  
    busca.ActiveConnection = cnd  
    busca.Open
```



```
regencontrados = busca.RecordCount

If regencontrados = 0 Then
Else
  For contador = 0 To regencontrados - 1
    codigo = busca("codigo")
    Combo3.AddItem codigo
    busca.MoveNext
  Next contador
End If
busca.Close
End Sub

Private Sub carga_combo_fallas()
  Combo4.Clear
  Sql = "SELECT * FROM tab_fallas ORDER BY descripcion"
  Set busca = New ADODB.Recordset
  busca.CursorLocation = adUseClient
  busca.CursorType = adOpenKeyset
  busca.LockType = adLockOptimistic
  busca.Source = Sql
  busca.ActiveConnection = cnd
  busca.Open
  regencontrados = busca.RecordCount

  If regencontrados = 0 Then
Else
  For contador = 0 To regencontrados - 1
    codigo = busca("codigo")
    Combo4.AddItem codigo
    busca.MoveNext
  Next contador
End If
busca.Close
End Sub

Private Sub carga_combo_referencia_carrier()
  Combo2.Clear
  Sql = "SELECT * FROM tab_referencia_carrier ORDER BY descripcion"
  Set busca = New ADODB.Recordset
  busca.CursorLocation = adUseClient
  busca.CursorType = adOpenKeyset
  busca.LockType = adLockOptimistic
  busca.Source = Sql
  busca.ActiveConnection = cnd
  busca.Open
```



```
regencontrados = busca.RecordCount

If regencontrados = 0 Then
Else
  For contador = 0 To regencontrados - 1
    referencia = busca("codigo") & " - " & busca("descripcion")
    Combo2.AddItem referencia
    busca.MoveNext
  Next contador
End If
busca.Close
End Sub
```

IV.3 Análisis, diseño y construcción del front-end.

IV.3.1 Diagrama Jerárquico de Funciones del Sistema.

Como ya se explicó en un capítulo anterior Visual Basic es un lenguaje de programación destinado a la creación, comprobación y depuración de aplicaciones.

Para elaborar el Sistema Integral de Control de Reportes, debemos necesariamente entender la forma en la que opera este lenguaje de programación y la forma en la que se estructura el desarrollo de una aplicación en el ambiente de Visual Basic.

El diagrama de la figura IV.3.1.1 esboza de manera general cada una de las etapas para el desarrollo y elaboración general una aplicación completa:

- **Inicio de Proyecto:** Se escoge el tipo de proyecto a elaborar, en este caso, estándar.
- **Integración de objetos a la forma del proyecto desde la barra de herramientas:** Se integran los objetos necesarios (botones, frames, listbox, combobox, etc) al área de trabajo de la forma del proyecto.



- **Modificar propiedades necesarias de los objetos integrados:** Se escogen y modifican cada una de las propiedades de los objetos para su correcta utilización.
- **Seleccionar evento al cual responderá cada objeto:** Por lo general en esta etapa se escogen los eventos al cual cada objeto responderá al ser llamado en la aplicación.
- **Programar evento de cada objeto integrado:** En esta etapa se tienen que programar detalladamente los eventos que se escogieron de cada objeto.
- **El código es ¿correcto?:** Etapa de toma de decisiones, el código es probado, es o no es correcta su funcionalidad.
- **Depurar eventos erróneos:** Si la anterior etapa arrojo errores de compilación el código es nuevamente ingresado a la programación de eventos.
- **Guardar proyecto:** Si el código no arrojo errores en la programación se guarda proyecto para pasar a la etapa final.
- **Generar proyecto ejecutable:** Si las etapas anteriores son satisfactorias se genera el proyecto final (exe).
- **Fin de proyecto:** Se concluye la elaboración de un proyecto.

Finalmente se redactan informes o manuales para su administración y/o utilización, así mismo, recomendaciones de uso para usuarios.

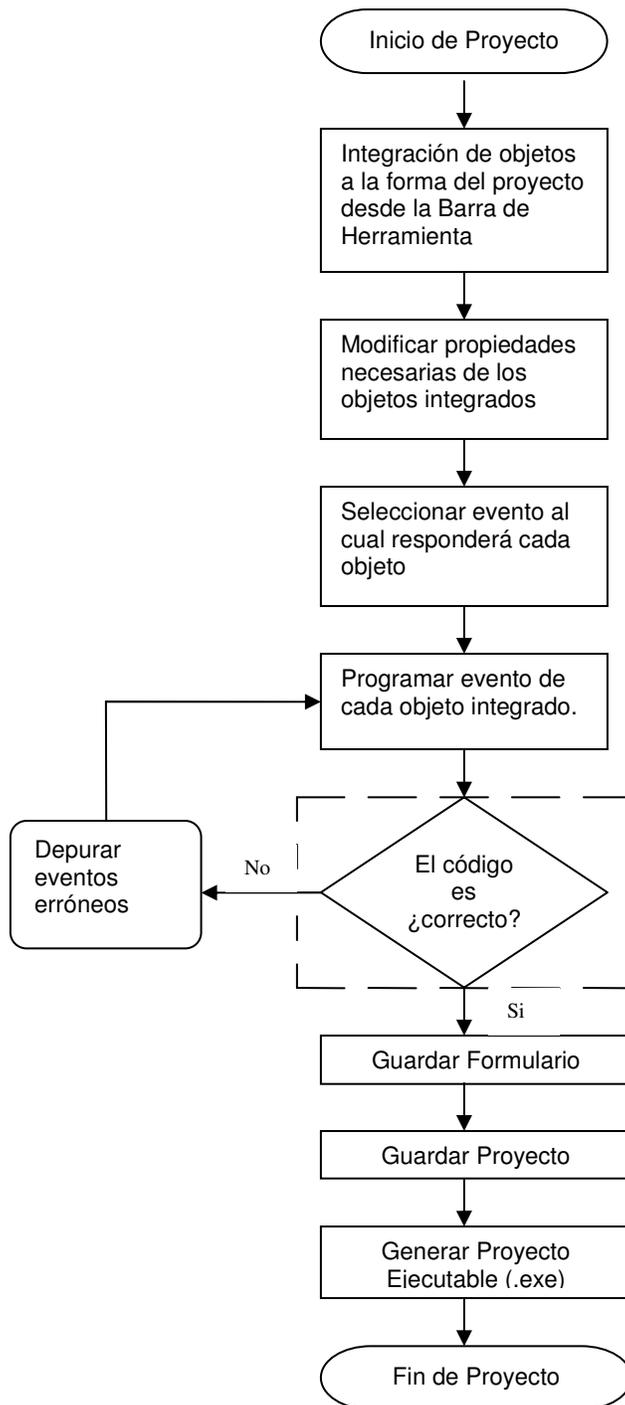


Figura IV.3.1.1 Diagrama de creación de un programa



IV.3.2 Desarrollo, prueba e integración de las diferentes pantallas con la base.

Es eficaz en la etapa de diseño, la integración de todos los controles a la Forma (form) del proyecto para tener un buen sistema funcional. El Sistema Integral de Control de Reportes para Celulares presenta las siguientes formas como interfaces de usuario.

- **Ingreso a SICRC.**



Figura IV.3.2.1 Pantalla Ingreso a SICRC

El objetivo principal y fundamental es dar acceso a SICRC validando que el usuario y password proporcionados sean los correctos,

La figura IV.3.2.1 muestra la pantalla de ingreso a SICRC, contiene los siguientes objetos:



Objeto	Función
Label	Muestra los mensajes: Usuario, Password y Versión 1.2.1
Textbox	Permite Ingresar los textos: usuario y contraseña

- **Menú Principal.**

Proporciona un menú interactivo y una barra de herramientas al usuario para ingresar a cada uno de los módulos independientes de SICRC: Administración, Clientes, Recepción, Entrega, Recibos, Laboratorio, Consultas Generales, Cambiar Password, Salir.



Figura IV.3.2.2 Menú Principal



La pantalla de la Figura IV 3.2.2 muestra los objetos que incluye el menú gráfico

Objeto	Función
CommandButton	Direcciona a cada módulo de SICRC

- **Módulo de Administración**

Permite llevar el control del Personal, Usuarios, Catálogos y algunos Parámetros del Sistema: como lo son parámetros de vencimiento de reportes y cargar archivos de Excel – Recepción y regresar al Menú Principal.

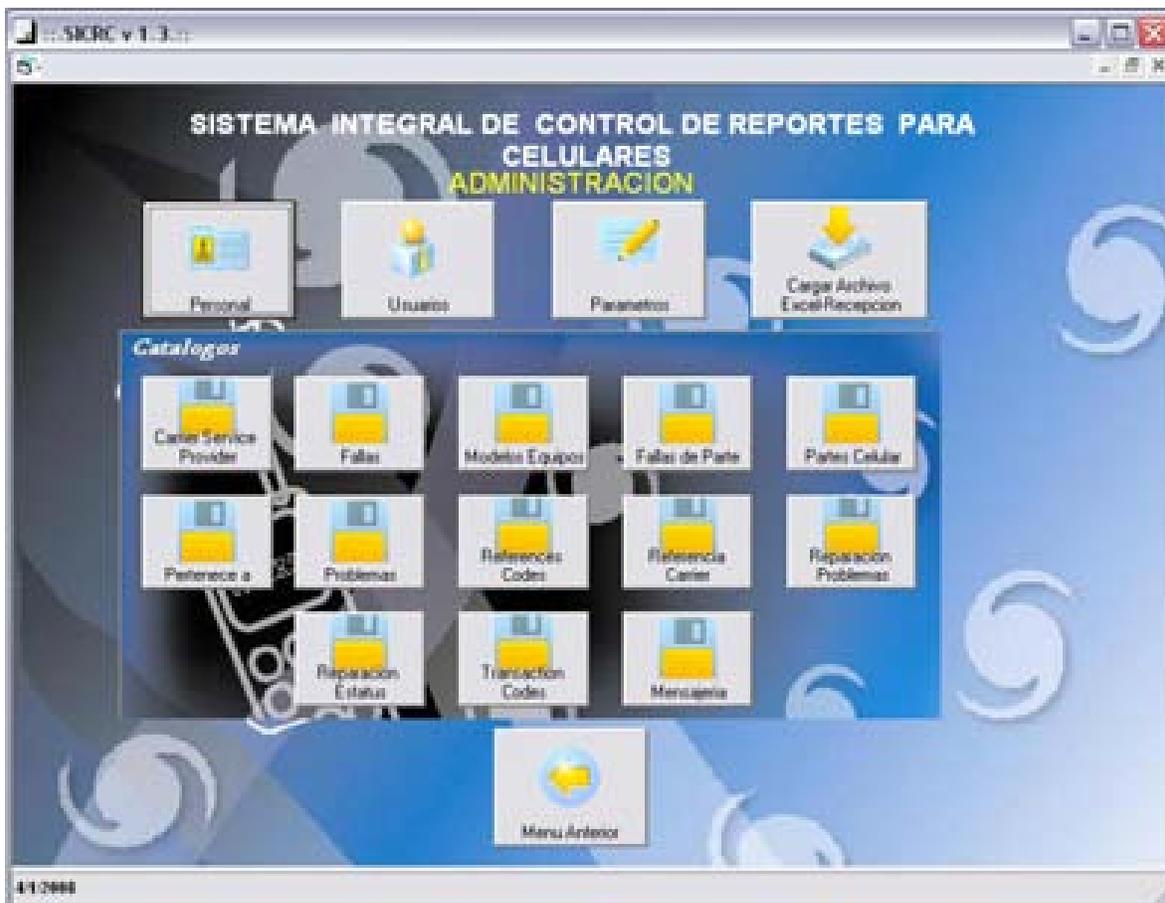


Figura IV 3.2.3 Menú Administración



La figura IV.3.2.3 muestra el contenido de la pantalla de administración, la cual contiene los objetos command button: Personal, Usuarios, Catálogos, Parámetros, Cargar Archivo Excel Recepción, Menú Anterior, las finalidades de estos son ingresar a las secciones de cada modulo.

- **Sección PERSONAL:** Da acceso a la sección de Administración de Personal (Figura IV.3.2.4) cuyo principal objetivo es dar de alta al personal que labora dentro de la empresa.

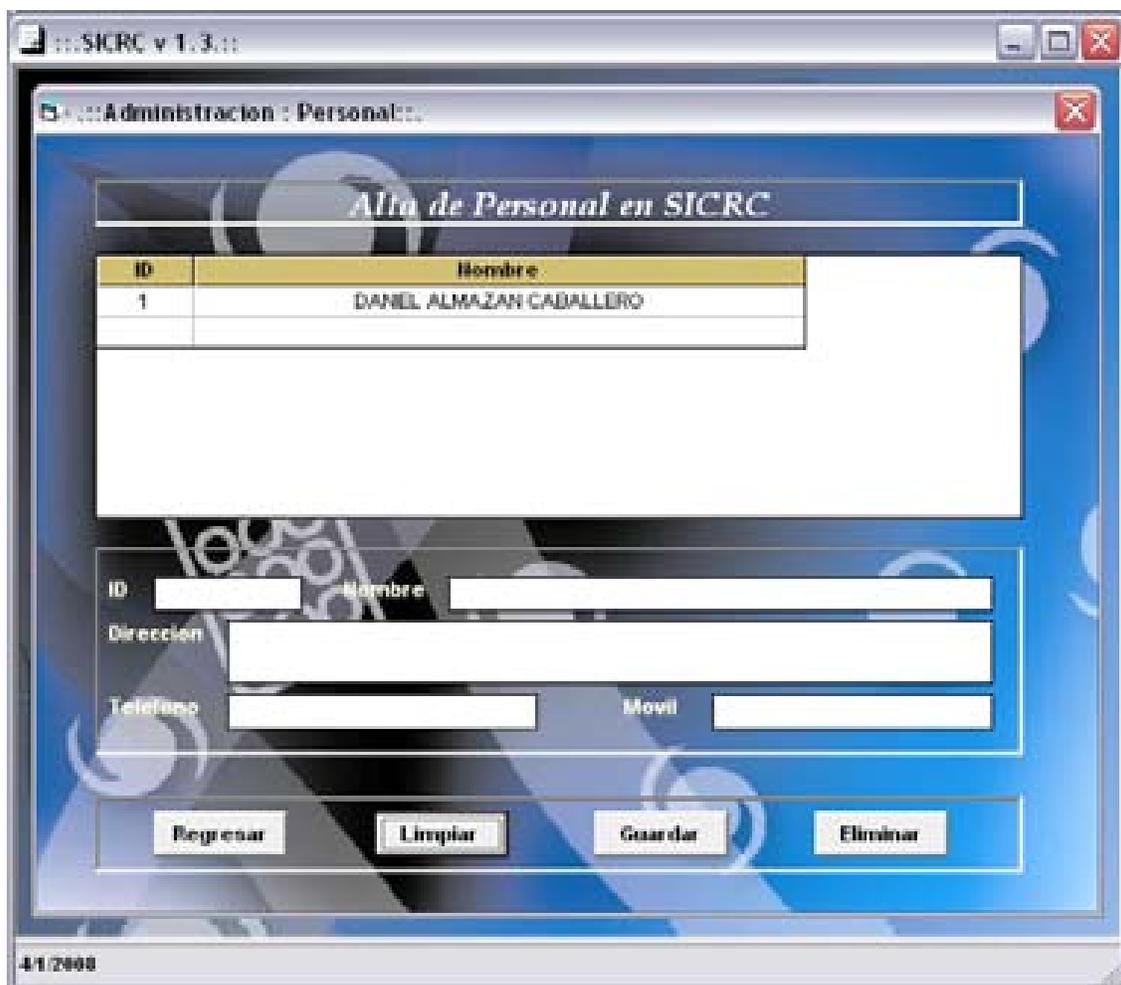


Figura IV.3.2.4 Sección Administración de Personal: Alta



La pantalla contiene los siguientes objetos:

Objeto	Función
MSFlexGrid	Muestra el numero total de columnas o filas (personal de la empresa) que existen en la tabla de Personal de la base de Datos SQL
Label	Muestra los mensajes: ID, Nombre, Dirección, Teléfono, Móvil.
TextBox	Permite la escritura de los datos de los labels anteriores, respectivamente.
CommandButton	<i>Cancelar:</i> Regresa al menú principal <i>Limpiar:</i> Limpia los textbox en caso de errores en el registro de alta. <i>Guardar:</i> Guarda el registro de alta en la base de datos SQL. <i>Eliminar:</i> Elimina un registro de la base de datos.

- **Sección USUARIOS:** Administra los accesos de los usuarios y da los permisos a las diferentes secciones de SICRC. La Figura IV.3.2.5 muestra la pantalla de Permisos a Usuarios

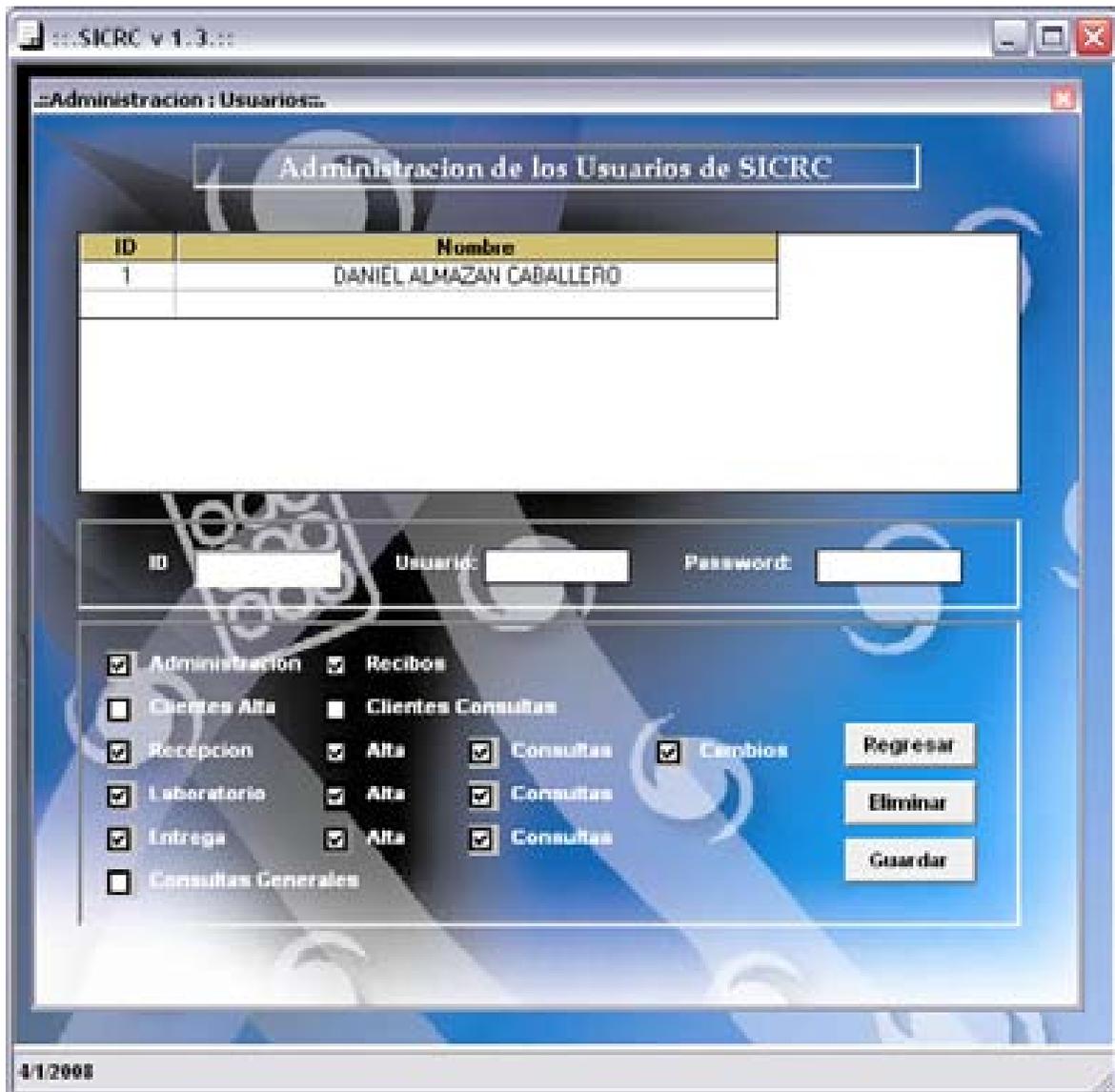


Figura IV.3.2.5 Control de permisos a usuarios



Esta pantalla contiene los siguientes objetos relevantes:

Objeto	Función
MSFlexGrid	Muestra el numero total de usuarios registrados en SICRC en la base de Datos SQL
CheckBox	Si se encuentran seleccionados todos se da acceso a los diferentes Módulos: <i>Administración, Recibos, Clientes, Recepción, Laboratorio, Entrega, Consultas Generales con sus respectivos permisos como: altas, cambios, consultas.</i> Si se encuentran seleccionados solo algunos se tiene exclusivamente permiso a ello, únicamente.

- **Sección PARÁMETROS:** Esta sección permite ingresar el número de días para determinar las órdenes de servicios vencidas en la empresa. (Figura IV.3.2.6)



Figura IV.3.2.6 Sección de Parámetros



- **Sección CARGAR ARCHIVO DE RECEPCIÓN:** Permite cargar desde un archivo de Excel el respaldo de la información contenida en la tabla de Recepción de la base de datos en caso de alguna incidencia mayor (daño a la tabla o daño a la base de datos). La ruta de la carga de archivo de recepción siempre estará ubicada en C:\Isatel\Recepcion\hoja1.xls

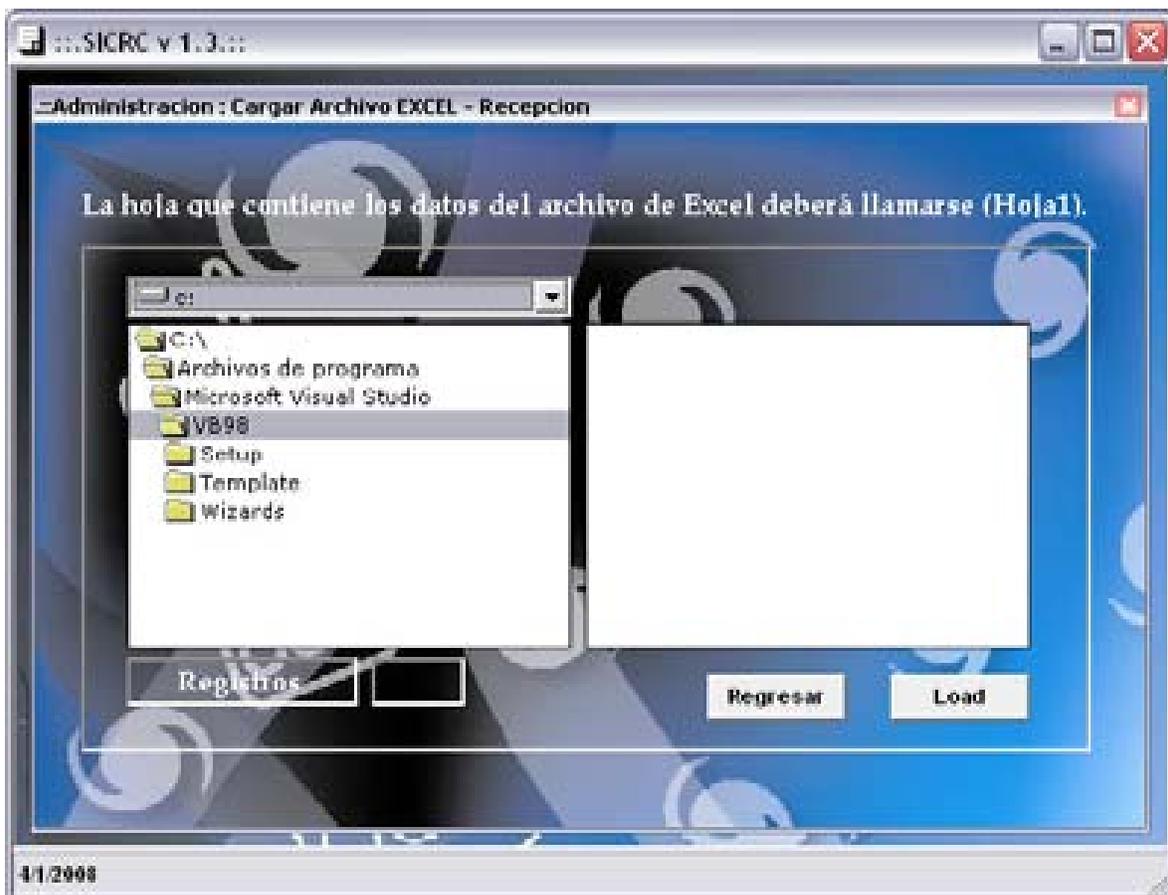


Figura IV.3.2.7 Sección Carga de Archivos

La Figura IV.3.2.7 contiene los siguientes objetos que comprenden la interfaz con el usuario. Algunos de los objetos nuevos son los siguientes:



Objeto	Función
DriveListBox	Permite ubicar la unidad de lectura en la cual se obtendrá la información de carga de archivo.
DirListBox	Muestra en lista el contenido de la unidad
FileListBox	Muestra en lista los archivos que se encuentran en el directorio.

- **Sección CATÁLOGOS:** Esta sección carga los catálogos que se muestran en la Figura IV 3.2.8, ingresando a cada una de esta opciones se direcciona a la carga del catálogo en el sistema SICRC.



Figura IV.3.2.8 Sección Catálogos

La pantalla de la Figura IV.3.2.9 muestra un ejemplo de carga del catálogo del proveedor.

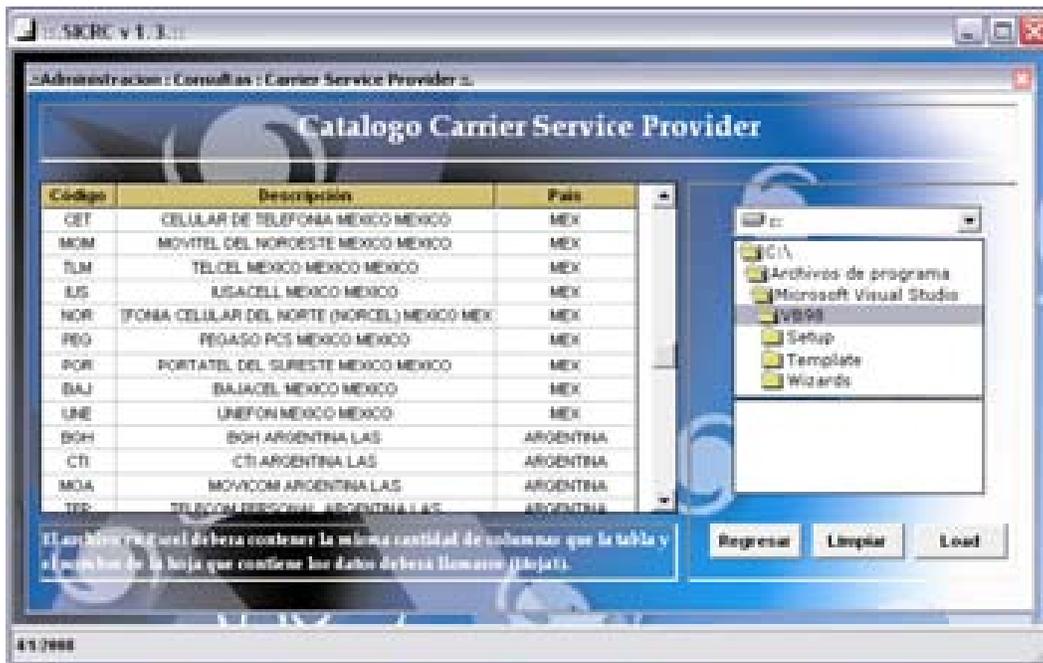


Figura IV.3.2.9 Sección de Carga de un Catalogo

Cada una de estas pantallas de catálogos contiene los mismos objetos que la pantalla Carga Archivo Recepción mencionada anteriormente.

- Módulo de Clientes.

Permite dar de alta a un nuevo cliente o empresa que requiera los servicios de la empresa, o bien, consultar los datos del cliente o empresa ya existentes en la base de datos. La Figura IV.3.2.10 muestra la pantalla de interfaz de usuario y contiene los siguientes objetos:

Objeto	Función
CommandButton	<p><i>Alta:</i> direcciona a la pantalla de sección de alta de clientes o empresas</p> <p><i>Consulta:</i> direcciona a la pantalla sección de consulta de clientes o empresas.</p>



Figura IV.3.2.10 Sección Clientes

- **Sección ALTA:** Da alta en un nuevo registro de la base de datos a un cliente o empresa (Figura IV.3.2.11). Esta pantalla contiene los siguientes objetos:

Objeto	Función
Labels	Muestra los mensajes de las etiquetas: Cliente, Empresa, Usuario que realiza la acción, fecha de modificación, etc.
TextBox	Permite la escritura de los datos para dar de alta al cliente o empresa: Idcliente, teléfono, cp, mail, fecha registro
CommandButton	<i>Cancelar:</i> Cancela el registro, sale de la sección



	de Clientes-Alta <i>Guardar:</i> Guarda el nuevo registro en la base de datos.
Frame	Separa los registros de alta por categorías: Cliente-Empresa, Dirección y Otros.

Alta de Clientes

Id Cliente: 0071

Datos:
(Paterno / Materno / Nombres)
Cliente: _____
Empresa: _____

Dirección:
Calle / Número: _____
Colonia: _____
Delegación / Municipio: _____
Estado: _____
Codigo Postal: _____

Otros:
Telefono: _____ R.F.C.: _____
Email: _____ Fecha Registro: 01/04/00

Guardar Regresar

Usuario: ADMIN Modificación: 01/04/00 15:30

412048

Figura IV 3.2.11 Alta de un Cliente o Empresa



- **Sección CONSULTA:** realiza la consulta del dato de un cliente o empresa registrado previamente en SICRC. Esta pantalla contiene únicamente un botón más que la pantalla de alta de clientes, su principal función es hacer la búsqueda del cliente en la base de datos (Figura IV 3.2.12), ya sea ingresando el id del cliente inmediatamente o presionando <Enter> para realizar la búsqueda manual,

SICRC v 1.3.00

Clientes : Consultas

Consulta de Clientes

Id Cliente **Buscar**

Datos
(Palermo / Malermo)

Nombre

Apellido

Dirección

Calle / Número

Colonia

Delegación / Municipio

Estado

Código Postal

Otros

Teléfono **R.F.C.**

Mail **Fecha Registro**

Guardar **Regresar**

Usuario ADMIN **Modificación** 01/04/08 15:38

41/2000

Figura IV 3.2.12 Consulta de Cliente o Empresa



Al utilizar esta opción de búsqueda manual se despliega una pantalla más la cual arroja los datos de todos los clientes o empresas (Figura IV.3.2.13).

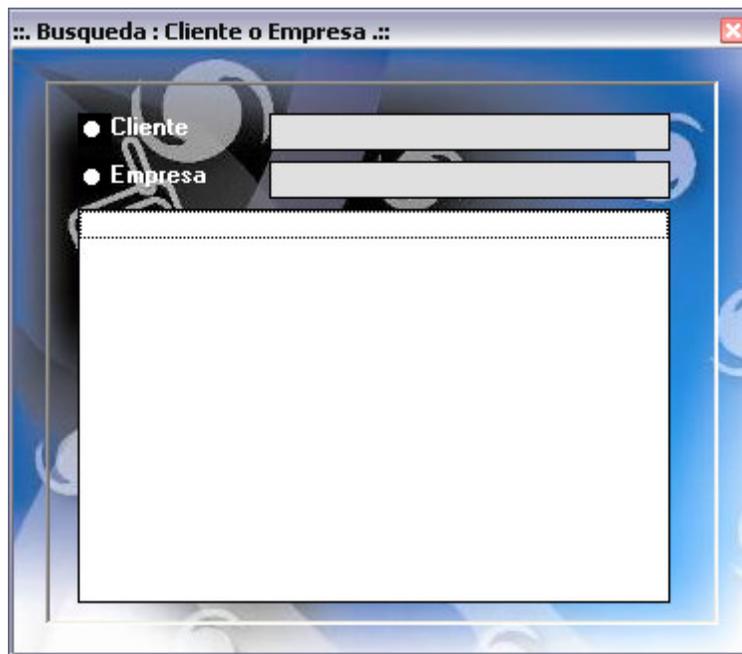


Figura IV.3.2.13 Búsqueda manual.

- Módulo de Recepción.

Este módulo nos permite realizar varias acciones, por ejemplo levantar un reporte de atención al cliente, es decir una nueva Orden de Servicio (captura), realizar cambios a esa orden de servicio previamente registrada en el sistema y en la base de datos, y además de realizar diversas consultas como lo muestra la Figura IV 3.2.14 en la sección de Consultas.

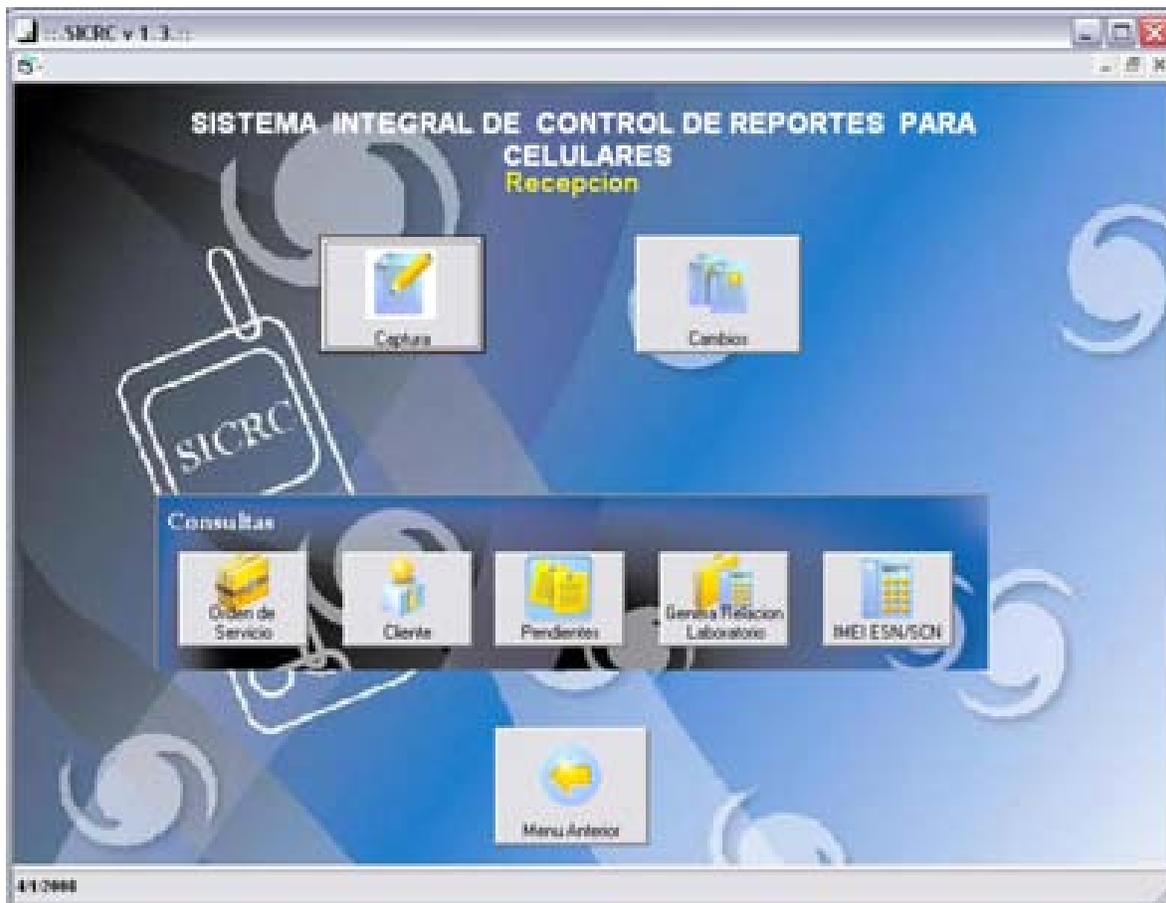


Figura IV 3.2.14 Modulo de Recepción.

Este módulo contiene los siguientes objetos: command button (7) como interfaz del usuario para ingresar a cada una de las secciones de Recepción.



- **Sección CAPTURA.** Esta sección como su nombre lo indica nos permite realizar una nueva captura de una orden de servicio para la empresa.

Figura IV.3.2.15

La Figura IV.3.2.15 muestra la pantalla de la nueva orden de Servicio, en ella encontramos los siguientes objetos:



Objeto	Función
Labels	Muestra los mensajes de las etiquetas: Orden Servicio, Folio Garantía, Fecha, Hora, Etiquetas de las Características del teléfono como Modelo, Tecnología, Códigos, Número de Minutos y Minutos Acumulados, etc., además de los accesorios que integran a ese celular, datos del cliente, etc.
TextBox	Permite la escritura de los datos para dar de alta la orden de servicio, folios de garantía, datos acerca del teléfono como códigos, versión del software, etc. y además, la escritura de datos del cliente como Id cliente, teléfono, cp, mail, fecha registro, etc.
CommandButton	<i>Cancelar:</i> Cancela el registro, sale de la sección de Recepción: Captura <i>Guardar:</i> Guarda el nuevo registro en la base de datos.
Combobox	Permite escoger alguna opción de un catálogo, en este ejemplo un código de referencia.
CheckBox	Permite seleccionar una o varias opciones para el registro de los accesorios del celular en la base de datos, ejemplo de ello: Goma, Cargador, Batería, etc.



La utilización de la sentencia Select tiene la siguiente estructura en el sistema:

*Select * from <tabla> where campo1=<parámetro de búsqueda>;*

Una vez encontrado el registro y haber llenado los campos previos de actualización, se ejecuta la sentencia de update con la siguiente estructura:

Update <tabla> set campo1=<parámetro de cambio1>,campo2=<parámetro de cambio2> where campo1=<registro de búsqueda>;

Con ello se garantiza que la Orden de Servicio se haya actualizado con los nuevos datos proporcionados.

- **Sección: CONSULTAS.** Esta sección nos permite realizar consultas a las Órdenes de Servicio generadas con anterioridad, consultar los datos de un cliente, las órdenes pendientes de atención, además de generar la relación del Laboratorio - si es que se va a quedar para alguna reparación el celular. La figura IV.3.2.17 muestra el catálogo de consultas en este módulo.



Figura IV.3.2.17 Catálogo del Módulo de Recepción.



- Consulta - Orden de Servicio. Permite visualizar el reporte de una Orden de Servicio previamente registrada. La Figura IV.3.2.17 muestra la pantalla de consulta de una orden de servicio ya registrada

La Figura IV.3.2.17 Orden de Servicio.

Los objetos de esta pantalla como interfaz de usuario son los mismos que la forma presentada en la sección de Captura.



- Consulta – Cliente: Permite consultar en el catálogo de clientes la orden de servicio de este cliente previamente registrado en el sistema, la búsqueda se puede hacer por cliente o empresa, la Figura IV.3.2.18 muestra la pantalla de consulta.

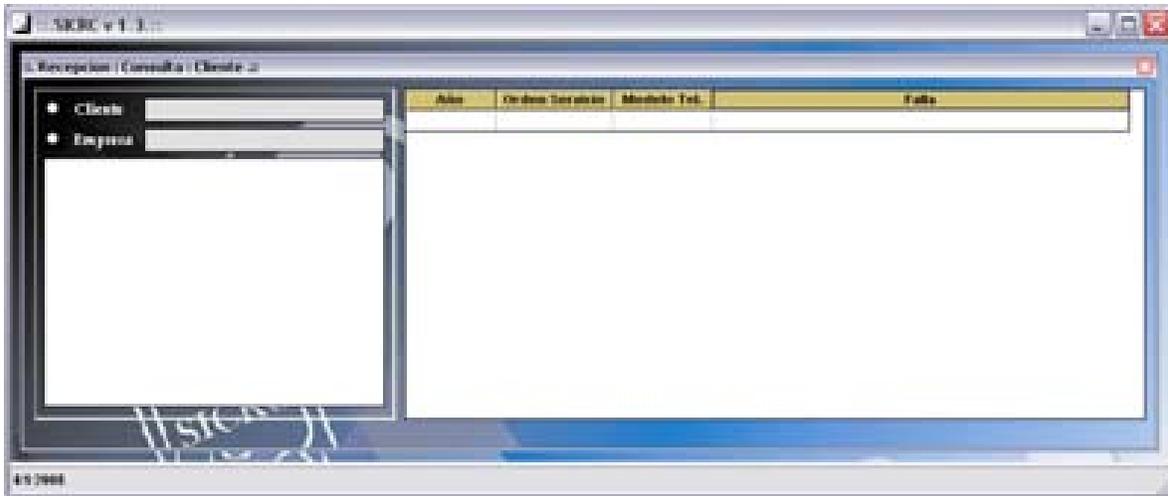


Figura IV.3.2.18

Los objetos de esta pantalla más importantes son:

Objeto	Función
MSFlexGrid	Muestra en un grid el numero de orden de servicio que fue levantado para el cliente arrojando el modelo del teléfono celular y la falla presentada por el dispositivo celular
ListBox	Muestra en forma de Lista los clientes o empresas registrados anteriormente.



- Consulta – Pendientes. Nos muestra todas las órdenes de servicio que aun se encuentran en pendiente de atención. (véase Figura IV.3.2.19)

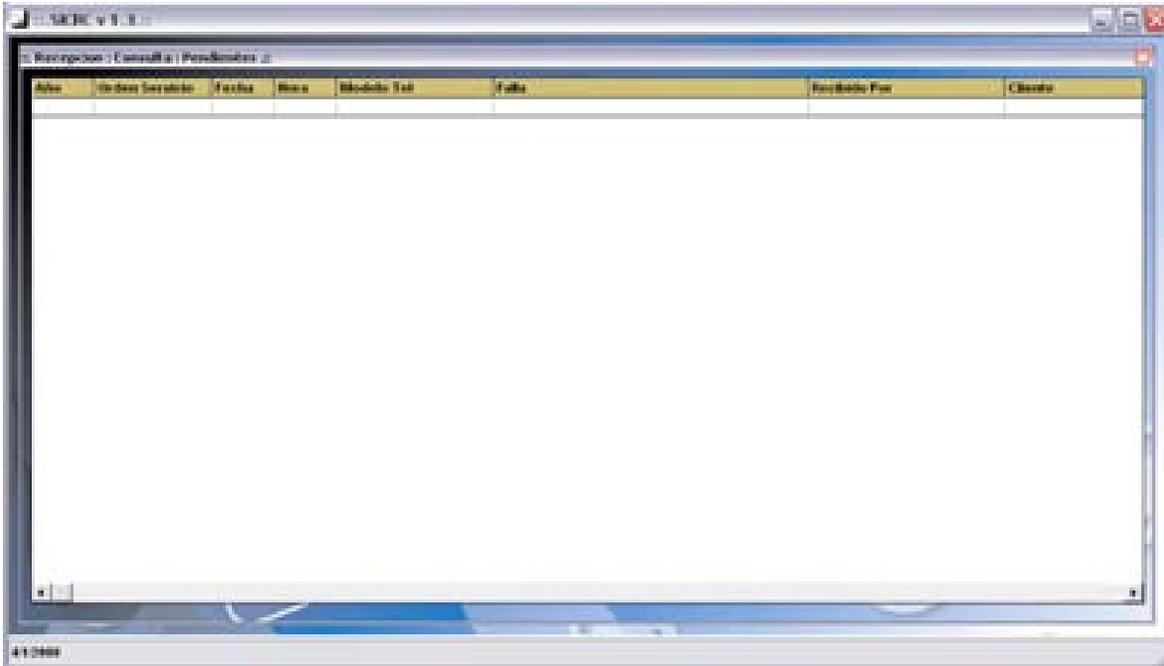


Figura IV.3.2.19 Pendientes de Atención

- Genera Relación Laboratorio. Esta sección permite generar la relación de atención a una orden de servicio para su atención en la Laboratorio.

La Figura IV.3.2.20 muestra la pantalla del sistema que procede a realizar la relación a Laboratorio, cuyos componentes más importantes son:

Objeto	Función
MSFlexGrid	Muestra en un grid el número de orden de servicio que fue levantado para el cliente y el cual se procederá a realizar su orden de servicio al Laboratorio.
Command Button	<i>Cancelar:</i> Cancela la generación de la orden de servicio. <i>Limpiar:</i> Realiza la limpieza de los MSFlexGrid <i>Generar:</i> Procede a realizar la Orden de Servicio al Laboratorio.

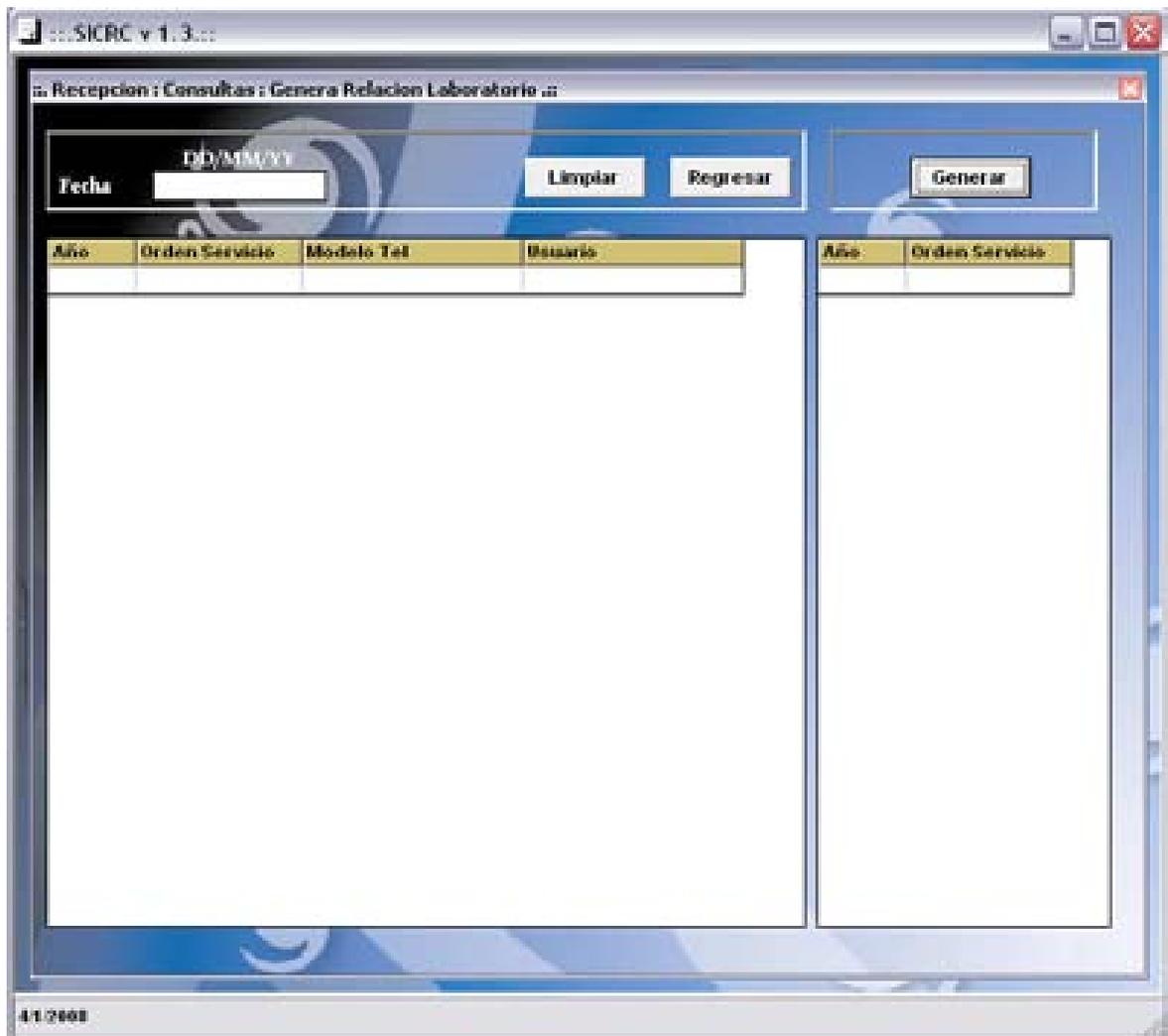


Figura IV.3.2.20 Genera la Relación de Laboratorio.

- Módulo de Entrega.

Este módulo nos permite realizar la captura de la información al momento de la entrega del celular y consultar las órdenes de servicios actuales, así como las pendientes como se observa en la Figura IV.3.2.21



Figura IV.3.2.21 Módulo de entrega

Este módulo contiene los siguientes objetos: command button (4) como interfaz del usuario para ingresar a cada una de las secciones de Entrega.

- **Sección CAPTURA.** Esta sección como su nombre lo indica nos permite realizar la captura de una orden de entrega para la empresa.

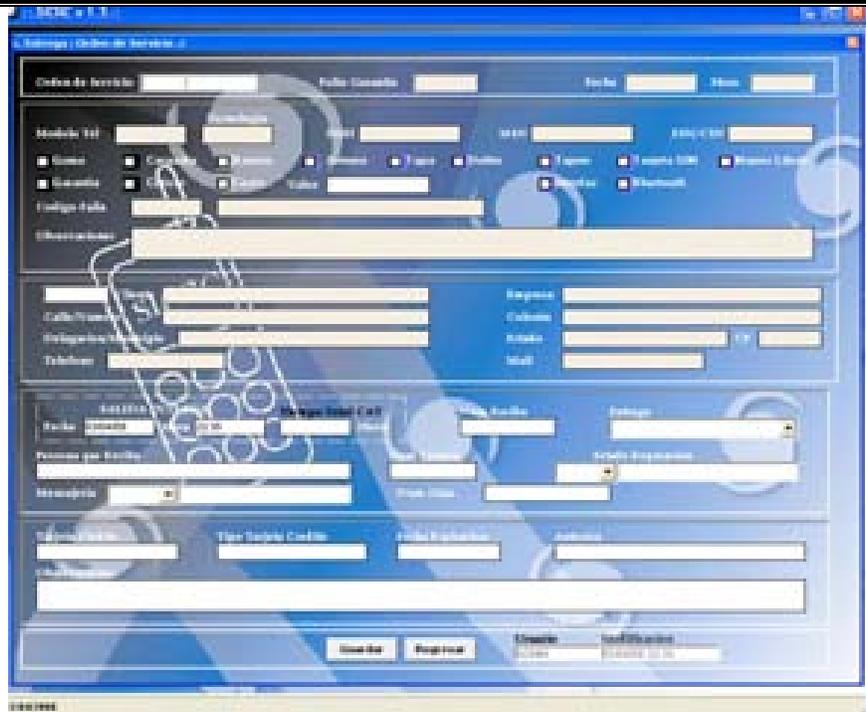


Figura IV.3.2.22 Pantalla de Captura de Entrega

La Figura IV.3.2.22 muestra la pantalla de la nueva orden de Entrega, en ella encontramos los siguientes objetos:

Esta tiene elementos semejantes a la Pantalla de Captura en la parte de recepción, pero solo nos permite la lectura de algunos campos y no la modificación directa. Algunos de los elementos distintos se muestran en la tabla siguiente.

Objeto	Función
TextBox	Permite la inserción de datos como son: la fecha y hora de la entrega, el número de recibo o factura que se le da al cliente, la persona que atendió. Se captura el numero de tarjeta, así como datos de la misma
TextArea	Se agregan comentarios sobre la entrega en caso de necesitarse
CommandButton	Verifica la información mostrada y guarda la entrega



- **Sección: CONSULTAS.** Esta sección nos permite realizar consultas de las órdenes de Servicio generadas con anterioridad, y los pendientes de entrega. La figura IV.3.2.23 muestra el catálogo de consultas en este módulo.



Figura IV.3.2.23 Catálogo del Módulo de Recepción.

- Consulta - Orden de Servicio. Permite visualizar el reporte de una Orden de Servicio previamente registrada. La Figura IV.3.2.24 muestra la pantalla de consulta de una orden de servicio ya registrada.

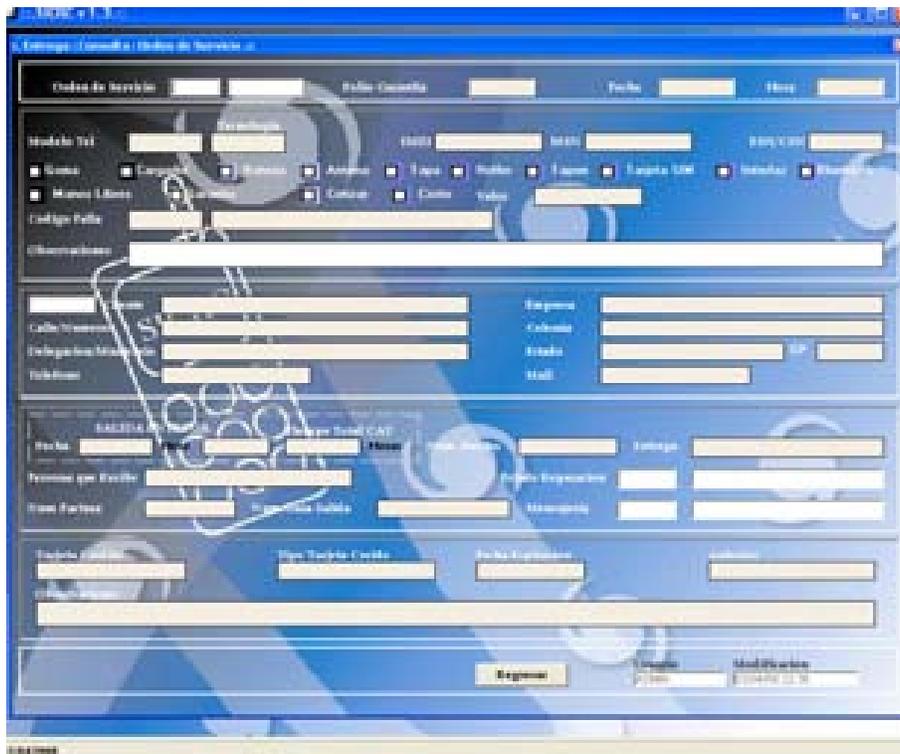


Figura IV.3.2.24 Pantalla de Consulta de Orden de Servicio a la Entrega



Los objetos de esta pantalla como interfaz de usuario son los mismos que la forma presentada en la sección de Captura.

- Pendientes. El objetivo de esta pantalla (Figura IV.3.2.25) conocer las ordenes que se tienen pendientes para la entrega de celulares, su forma es parecida a la pantalla de consulta general de laboratorio.

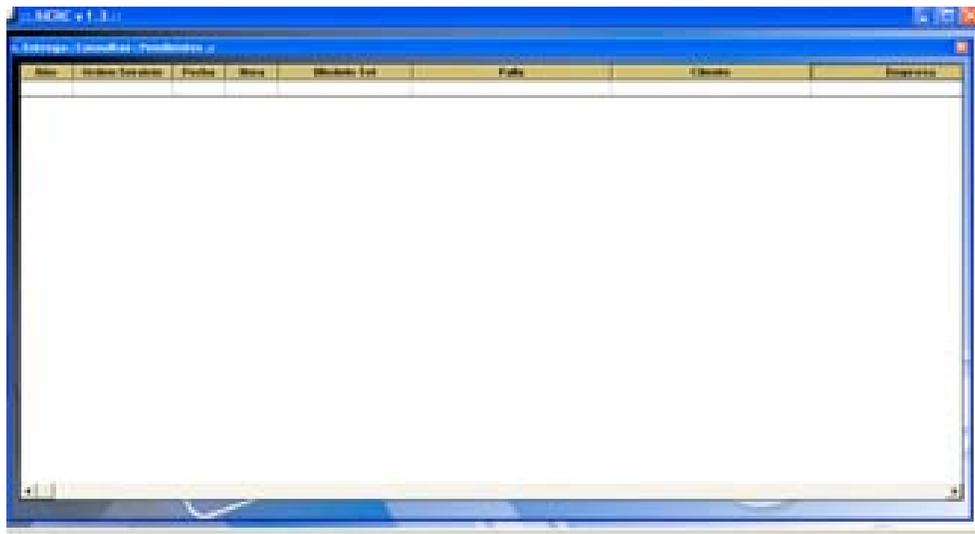


Figura IV.3.2.25 Pantalla de Consulta de Ordenes Pendientes de Entrega.

- Módulo de Recibos.

Este módulo nos permite imprimir los recibos de entrada y de pago para la entrega del comprobante del servicio realizado al celular Figura IV.3.2.26.

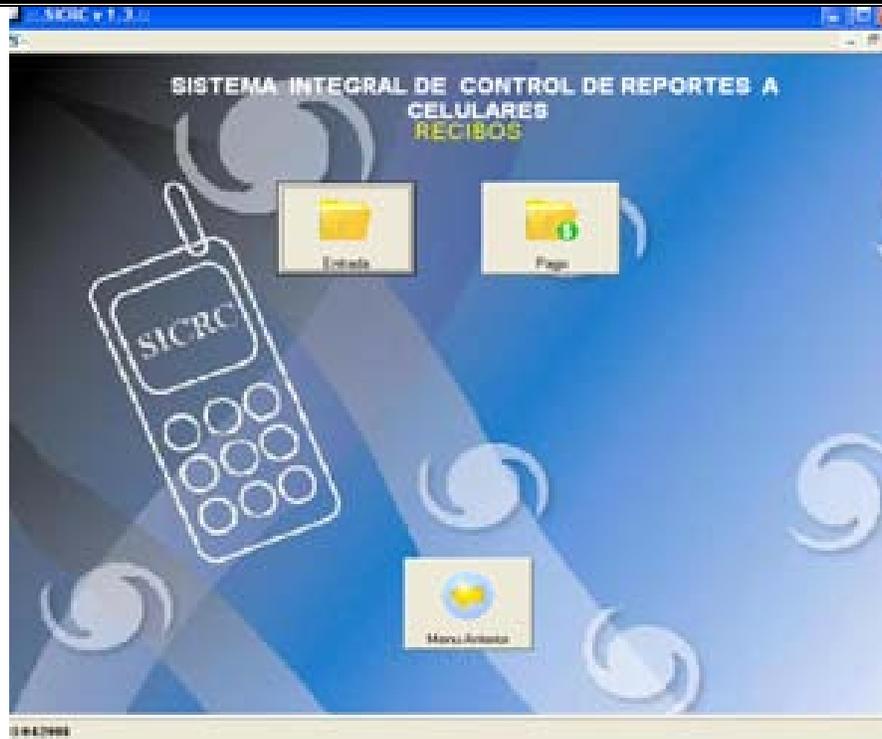


Figura IV.3.2.26 módulo de Recibos

- Sección - Recibos de Entrada

El funcionamiento de esta pantalla (Figura IV.3.2.27) radica en imprimir mediante un archivo de Excel la información relacionada con los recibos entrada al momento de recibir el celular, esto se hace con un FileListBox para seleccionar el archivo y el botón de imprimir que realiza el proceso.

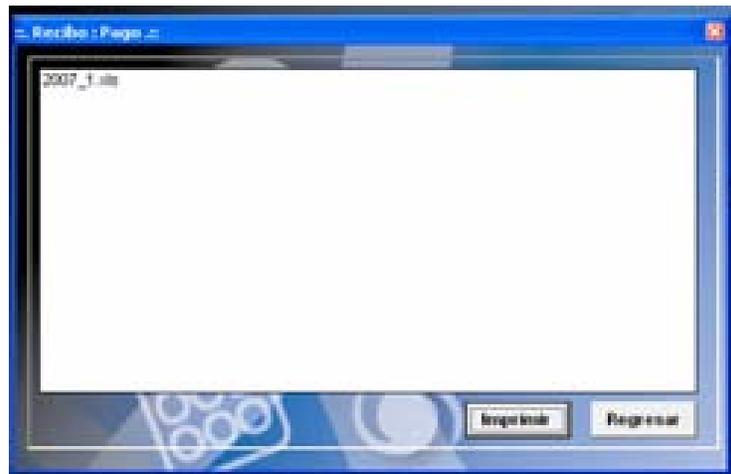


Pantalla de Recibos de Entrada (Figura IV.3.2.27)



- Sección - Recibos de Pago

Su meta es imprimir la información necesaria para un recibo de pago al momento de terminar la reparación del celular a un archivo de Excel, su forma es semejante a la pantalla anteriormente mencionada.



Pantalla de Recibos de Entrada (Figura IV.3.2.28)

- Módulo de Laboratorio.

Este módulo nos permite ingresar la información necesaria para el laboratorio así como realizar las consultas necesarias Figura IV.3.2.28



Figura IV.3.2.28 Módulo de Laboratorio



- Sección - Captura

La función de esta pantalla (Figura IV.3.2.29) es obtener la información necesaria del problema que presenta el teléfono, la persona que realiza la verificación, los códigos de reparación , el estado del equipo, los problemas encontrados así como su reparación correspondiente y las partes que necesiten cambiar.

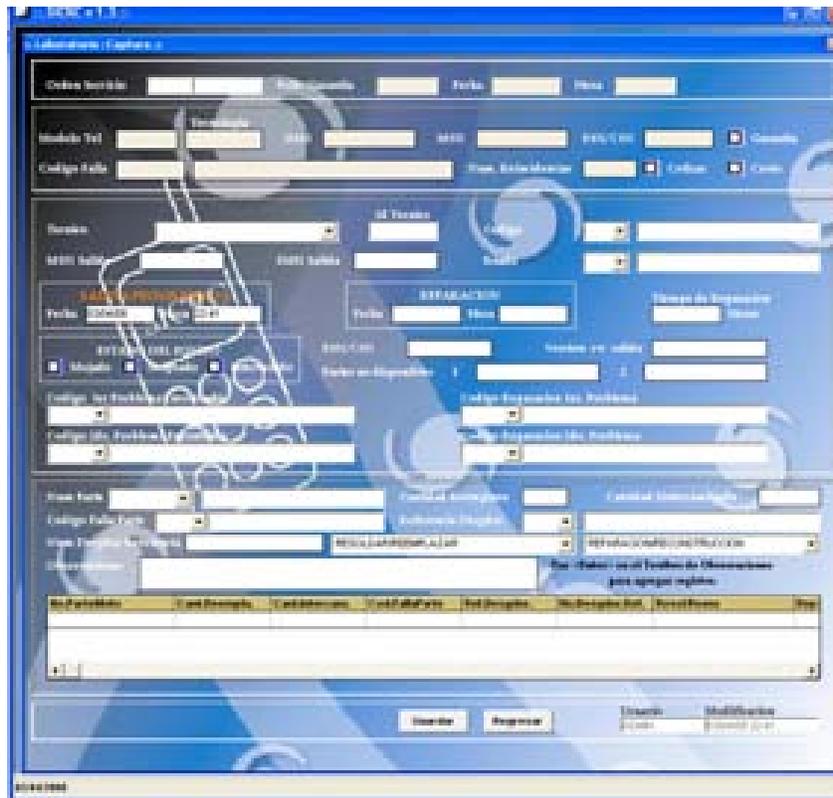


Figura IV.3.2.29 Pantalla de Captura de Información para el laboratorio

Objeto	Función
Label	Muestra un texto al usuario para conocer la información que se le solicita
TextBox	Se ingresa la información de garantía, costo, cotizar, mojado, golpeado e intervenido



Checkbox	Indica el estado del equipo
Combobox	Muestra una lista para poder seleccionar el elemento requerido por el usuario, ejemplo de esto son: Técnico, código de reparación, estado, número de parte
MSFlexGrid	Muestra la información en un arreglo de celdas de las distintas fallas.
ComandButton	Se encarga de verificar que parte de la información mostrada en la pantalla o se cancela la acción

- Sección - Consulta de orden de servicio del laboratorio

El funcionamiento de esta pantalla (Figura IV.3.2.30) es muy similar a la anteriormente mostrada, pero el fin de ella es meramente informativo, ya que solo permite mostrar la información capturada con anterioridad.

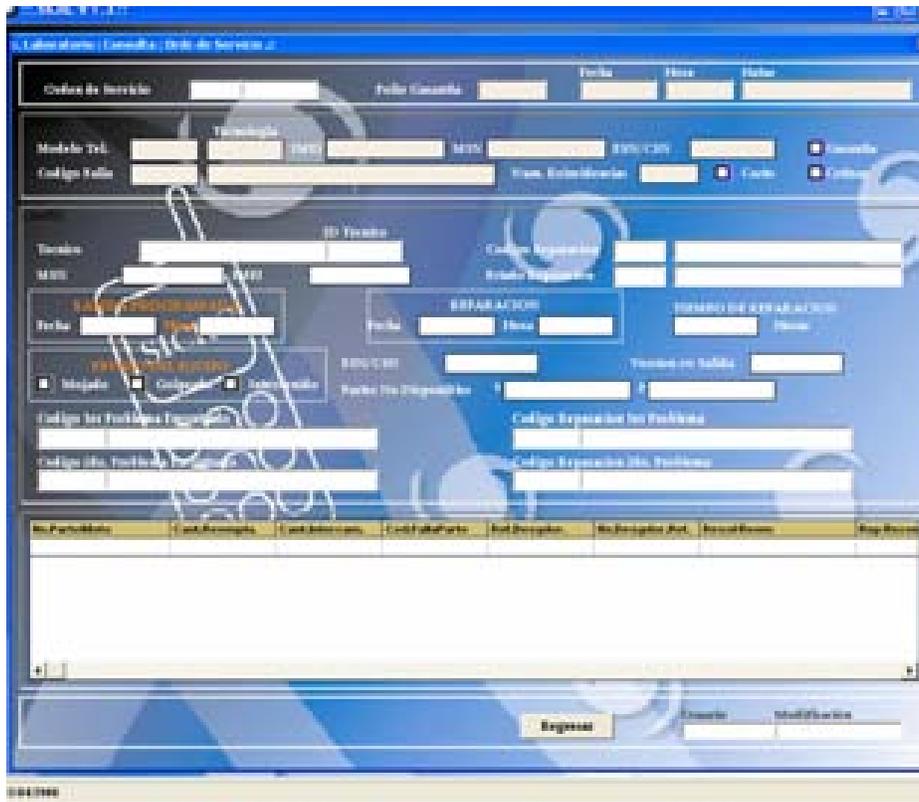


Figura IV.3.2.30 Pantalla de Consulta de orden de servicio del laboratorio



- Sección - Consulta General de Laboratorio

Esta pantalla (Figura IV.3.2.31) contiene los mismos elementos que la pantalla de atendidos por empleado solo que la información que se muestra en esta pantalla se refiere a las ordenes de laboratorio que han sido atendidas.

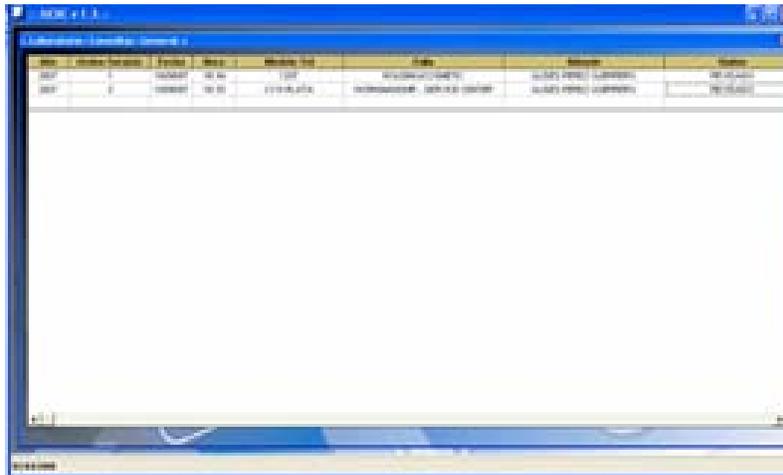


Figura IV.3.2.31 Pantalla de Consulta General de Laboratorio

- Sección - Generar relación salida de laboratorio

Esta pantalla (Figura IV.3.2.32) genera de acuerdo a una fecha, las órdenes de servicio que se han atendido por el laboratorio, para poder grabarlas si se requiere esta información en un archivo de Excel.

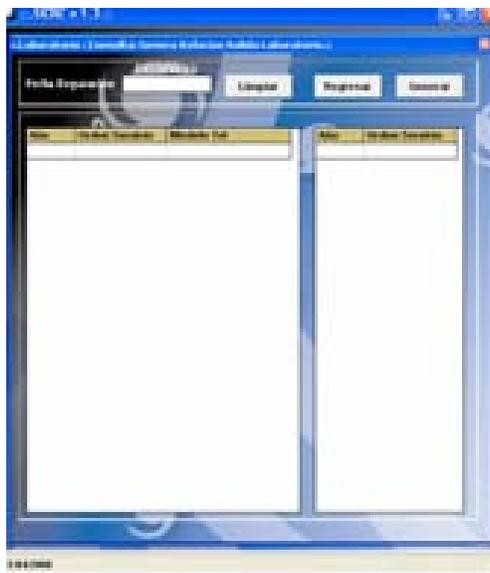


Figura IV.3.2.32 Pantalla Generar relación salida de laboratorio



Objeto	Función
Label	Muestra información al usuario como es, la fecha de reparación
TextBox	Captura la fecha de reparación para generar la consulta
CommandButton	Elimina el contenido del cuadro de texto y Generar manda la información de salida del laboratorio a un archivo de Excel
MSGlexFrid	Muestra en un gris, el Año, Orden de Servicio y Modelo del teléfono

- Módulo de Consultas Generales.

Este modulo nos permite buscar información de las ordenes pendientes de atención, las que han sido atendidas por los empleados y buscar un celular por el imei Figura IV.3.2.33



Figura IV.3.2.33 Módulo de Consultas Generales.



- Sección - Pendientes de Atención

El objetivo general de esta pantalla (Figura IV.3.2.34) es conocer con detalle las ordenes que están pendientes por atender mostrando información de cada una de ellas en forma detallada

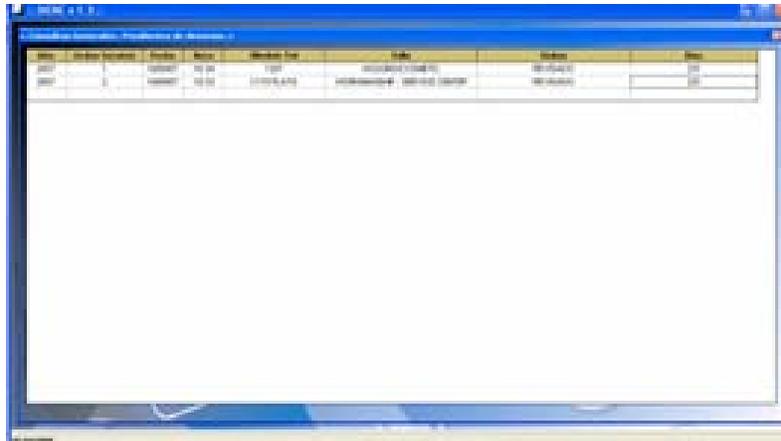


Figura IV.3.2.34 Pantalla Órdenes pendientes de Atención

Objeto	Función
MSFlexGrid	Muestra la información en un arreglo de celdas con los encabezados Año, Orden de Servicio, Fecha, hora, Modelo del Teléfono, Falla, Status y Días

Al entrar en esta pantalla se realiza la siguiente consulta SQL

`SELECT * FROM tab_recepcion WHERE status NOT LIKE "" & status_paso & ""`,
 cuya función es buscar dentro de la tabla de recepción aquellas ordenes cuyo estado de paso este vacío.

- Sección - Atendidos por empleado

La función de esta pantalla (Figura IV.3.2.35) es conocer los casos que se ha atendido cada uno de los empleados. Esto nos da un histórico del rendimiento de los empleados, conociendo los principales elementos que al cliente le interesan.



Figura IV.3.2.35 Pantalla de Atendidos por empleado

Objeto	Función
MSFlexGrid	Muestra la información en un arreglo de celdas con los encabezados Año, Orden de Servicio, Fecha, hora, Modelo del Teléfono, Falla, Recibido y Atendido

Al entrar en esta pantalla se realiza la siguiente consulta SQL

```
SELECT DISTINCT año,orden_servicio,idpersonal FROM tab_laboratorio WHERE status LIKE "" & status_paso & """, cuya función es buscar dentro de la tabla de laboratorio aquellas ordenes cuyo estado de paso este vacío.
```

- Sección - Consulta General por IMEI ESN/SCN

La funcionalidad de esta pantalla (Figura IV.3.2.36) radica en buscar por el número de IMEI o por el número de ESN/SCN un celular que anteriormente ha entrado al sistema contando así con un histórico del mismo.

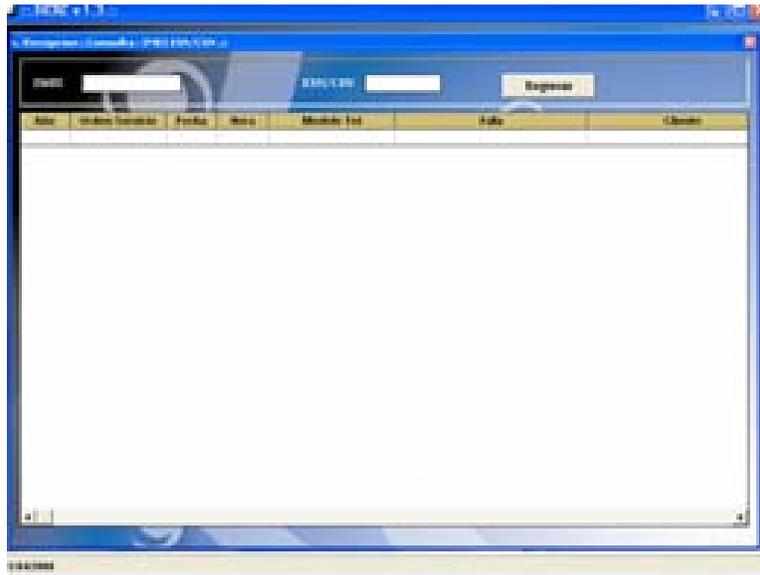


Figura IV.3.2.36 Pantalla de Consulta General por IMEI ESN/SCN

Objeto	Función
Label	Muestra un texto al usuario como imei y Esn/scn
TextBox	El usuario coloca información del imei o el Esn
MSFlexGrid	Muestra la información en un arreglo de celdas con los encabezados Año, Orden de Servicio, Fecha, hora, Modelo del Teléfono, Falla, Recibido y Atendido

Al entrar en esta pantalla se realiza la siguiente consulta SQL

"SELECT * FROM tab_recepcion WHERE imei LIKE '" & imei_paso & "'", esta consulta toma todos los campos de la tabla de recepción con el imei ingresado en la pantalla.

- Modulo de Modificación de Contraseña

El objetivo general de esta pantalla (Figura IV.3.2.37) es modificar la contraseña que tiene el usuario actual por uno nuevo por cuestiones de seguridad.



Figura IV.3.2.37 Pantalla de Cambio de Password

La pantalla contiene los siguientes elementos:

Objeto	Función
Frame	Enmarca los diversos objetos en pantalla
Label	Muestra un texto al usuario como: Password actual Nuevo Password, Confirmar Password
Textbox	Captura la información
Commandbutton	Guardar - Verifica en una primera instancia que los tres cuadros de texto tengan información si la información es correcta busca al usuario mediante una búsqueda SQL a través de ADO, si los datos son correctos actualiza la contraseña Cancelar – Sale de la ventana

- Botón de Salida

Nos permite salir de la aplicación para regresar el entorno del sistema operativo

Figura IV.3.2.38



Figura IV.3.2.38 Botón de Salir



IV.4 Pruebas, Integración y Afinación del Sistema.

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

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

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

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

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

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

- Pruebas de caja negra
- Pruebas de caja blanca
- Pruebas de integración
- Pruebas de rendimiento



- Pruebas de regresión
- Pruebas de volumen
- Pruebas de seguridad
- Pruebas de aceptación
- Pruebas alfa
- Pruebas beta

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

Pruebas de Caja Negra (Funcional)

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

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



Pruebas de Caja Blanca (Estructural)

Este tipo de pruebas son el contraste al de la caja negra, debido a que quién realiza la prueba puede analizar el código, estas pruebas son mucho mas amplias, normalmente se denominan pruebas de cobertura o pruebas de caja transparente, al total de pruebas de caja blanca se le llama cobertura, la cobertura es un número porcentual que indica cuanto código del programa se ha probado. Este tipo de prueba se realiza cuando se desarrolla un sistema, debido a que el programador conoce la lógica del sistema y la estructura de un componente para poder obtener los datos de prueba, conforme va realizando cualquier adecuación o desarrollo se corren pruebas para verificar la información, en muchas ocasiones, estas pruebas se realizan con ayuda de un usuario responsable del módulo del sistema en general, buscando:

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

Pruebas de Integración

Estas pruebas se realizan a un grupo de programas para asegurar que los datos u controles sean pasados adecuadamente entre ellos mismos. La prueba de integración es una técnica sistemática para construir la estructura del programa mientras que, al mismo tiempo, se llevan a cabo pruebas para detectar errores asociados a la interacción. Los tipos fundamentales de integración son:

- **Integración incremental.** Se combina el siguiente modulo que se debe probar con el conjunto de módulos que ya han sido probados.
- **Integración no incremental.** Se prueba cada módulo por separado y luego se integran todos de una vez y se prueba el sistema completo.



- **Ascendente.** Se comienza por los módulos hoja o base.
- **Descendente.** Se comienza por el módulo raíz y se va expandiendo hacia abajo.

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

La integración contempla aspectos como:

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

Pruebas de Regresión

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

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



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

Pruebas de Volumen

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

Pruebas de Seguridad

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

Pruebas de Aceptación

Estas pruebas las realiza el cliente. Son básicamente pruebas funcionales sobre el sistema completo y buscan una cobertura de la especificación de requisitos. Estas pruebas no se realizan durante el desarrollo, ya que sería impresentable de cara al cliente, sino una vez realizadas todas las pruebas de integración por parte del desarrollador.

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

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



Pruebas Alfa y Beta.

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

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

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

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

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



Pruebas Aplicadas al Sistema.

Para la validación de usuarios se intentó dejar los campos de Usuario y Password vacíos, pero el sistema no lo permite, permaneciendo la pantalla de acceso al sistema hasta en tanto no se ingrese algún carácter en ambos campos. Ver figura IV.4.1.



Figura IV.4.1. Pantalla para ingresar al sistema.

Posteriormente se ingresaron datos no validos y el sistema no nos da acceso. Ver figura IV.4.2.

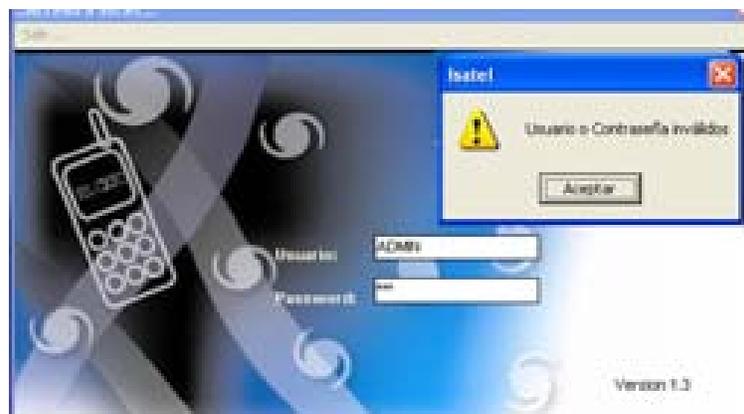


Figura IV.4.2 Ingreso de usuario o password incorrectos, el sistema envía mensaje de valores incorrectos.



Las pruebas de validación proporcionan una seguridad final de que el software satisface todos los requisitos funcionales de comportamiento y rendimiento.

De las pruebas que se hicieron para la validación de datos fue la de introducir en el módulo de administración en la opción de parámetros, un campo vacío y el sistema nos indica que se debe capturar un valor. Ver figura IV.4.3.

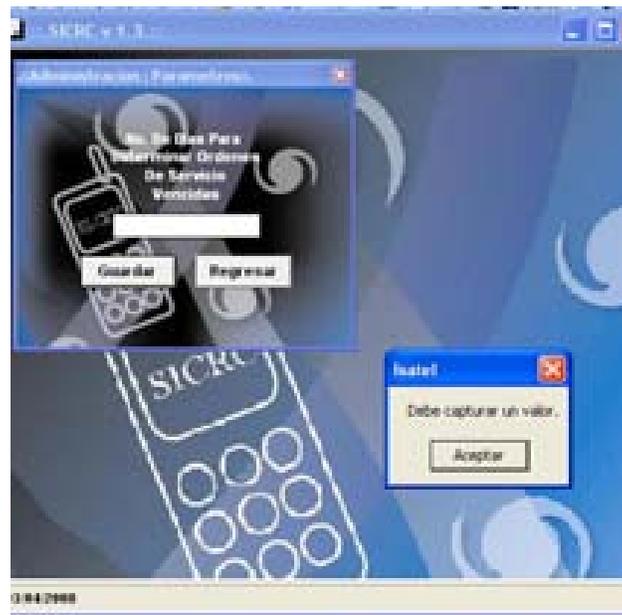


Figura IV.4.3. No permite el sistema ingresar campos en blanco.

En el módulo de administración en la opción de altas de personal, el sistema no permite introducir campos vacíos, nos indica que se debe capturar un valor. Ver figura IV.4.3.



Figura IV.4.3. No permite el sistema ingresar campos en blanco.

En el módulo de administración para dar privilegios a los usuarios el sistema no permite dar campos nulos para un usuario. Ver figura IV.4.4.



Figura IV.4.4. Se debe seleccionar un empleado para la asignación de privilegios.



Una de las fallas del sistema se detectó en el módulo de laboratorio en la opción de consultas, al momento de generar una relación de salida del laboratorio, el campo de fecha de reparación acepta valores que no son de fecha, pero esto hace que el sistema falle al encontrar dicha condición, se realizó la corrección correspondiente, Ver figuras IV.4.7. y IV.4.8.

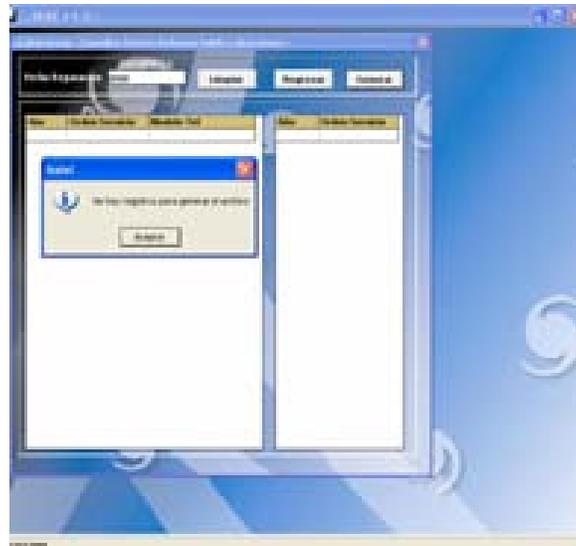


Figura IV.4.8. Falla del sistema al encontrar un campo con dato nulo.

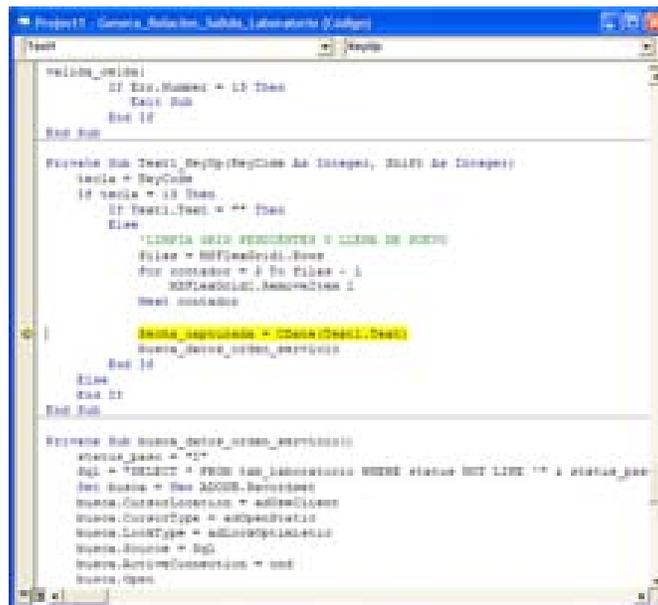


Figura IV.4.9. Al depurar el código muestra la sentencia que provoca la falla.



En el módulo de recepción, en la opción de captura de una orden de servicio, el sistema detecta como error al querer guardar sin haber introducido todos los datos que no acepta valores nulos. Ver figura IV.4.10.

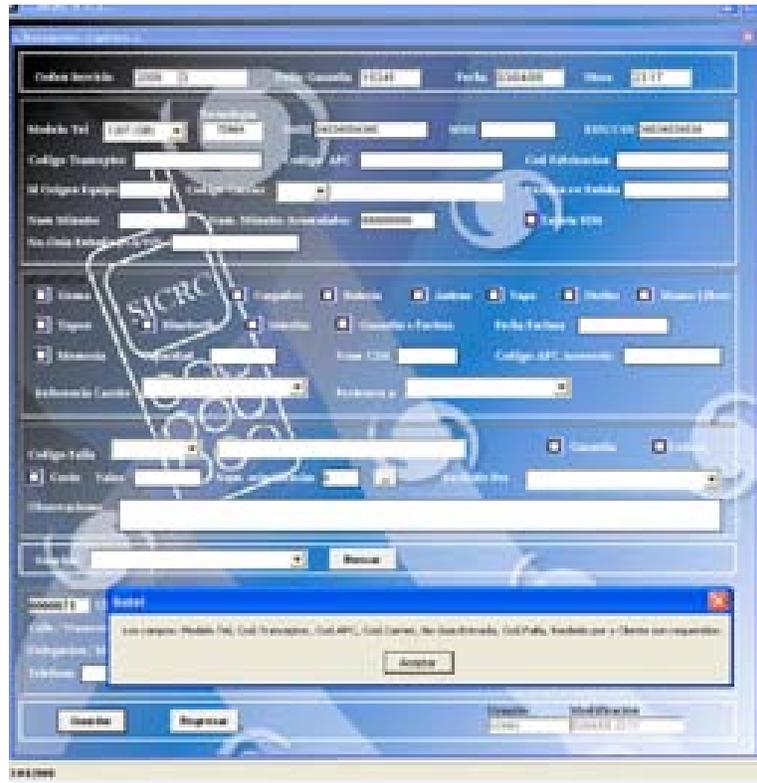


Figura IV.4.10. Error al faltar introducir datos en los campos.

IV.5 Mantenimiento del sistema y liberación

IV.5 Mantenimiento del Sistema

En esta sección, describiremos cuatro tipos de mantenimiento:

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



IV.5.1 Mantenimiento Preventivo

En este tipo de mantenimiento se previenen errores. Se implementa cuando se está desarrollando el software para evitar a futuro proporcionar un mantenimiento mayor y garantizar que los resultados dados por el sistema sean los correctos.

IV.5.1.1 Mantenimiento Correctivo

La primera actividad del mantenimiento se da ocasionalmente cuando la prueba del software no haya descubierto todos los errores latentes de un sistema. Durante el uso del sistema se encontrarán errores, los cuales deben ser informados al equipo de desarrollo. El proceso que incluye el diagnóstico y corrección de uno o más errores se denomina mantenimiento correctivo.

IV.5.1.2 Mantenimiento Perfectivo

Esta actividad de mantenimiento se da cuando un paquete de software tiene éxito. A medida que se usa el software, se reciben de los usuarios recomendaciones sobre nuevas posibilidades acerca de modificaciones a funciones ya existentes. Para satisfacer estas peticiones se lleva a cabo el mantenimiento perfectivo.

El mantenimiento perfectivo comprende también los cambios solicitados al programador del sistema, para modificar u optimizar algún módulo.

IV.5.1.3 Mantenimiento Adaptativo

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

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



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

- **Mantenimiento Aumentativo**

Este tipo de mantenimiento se da cuando se incluyen nuevas funciones que no se contemplan al inicio del desarrollo del sistema y surgen como una necesidad del usuario.

- **Mantenimiento Tecnológico**

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

La mayor parte del mantenimiento de adaptaciones se hace como respuesta a la aparición de nuevos problemas de la empresa, nuevas necesidades de información o nuevas ideas de mejoras. Estas actividades reciben el nombre de mejoras del sistema. El objetivo de las mejoras al sistema es modificar o ampliar el sistema de aplicaciones como respuesta a las necesidades cambiantes de la empresa. Este objetivo puede relacionarse con los bloques elementales de los sistemas de información del modo siguiente:

- **Personas:**

En su mayoría, las mejoras a los sistemas son propuestas por los usuarios de los sistemas, si bien los analistas, diseñadores y constructores de sistemas también pueden detectar posibles problemas técnicos relativos al rendimiento, la seguridad y los controles internos.



- **Datos:**

Muchas mejoras de los sistemas son demandas de nueva información que pueden derivarse de datos almacenados existentes. Algunas mejoras de datos pueden requerir la ampliación del almacenamiento de estos.

- **Procesos:**

En su mayoría, las mejoras a los sistemas requieren la modificación de programas existentes o la creación de nuevos programas para ampliar el ámbito general de sistemas de aplicaciones.

- **Redes:**

Al tener medios de comunicación más rápidos, eficientes y precisos, el sistema puede ser capaz de realizar nuevas tareas más pesadas y complejas.

- **Tecnología:**

En su mayoría, las mejoras a los sistemas se basan en ésta.

Los anteriores ejemplos de mantenimiento de adaptaciones se encuentran dentro de la categoría de re-ingeniería. Los objetivos de la re-ingeniería son: o bien adaptar el sistema ante un cambio tecnológico importante y arreglar el sistema antes de que falle o bien hacer el sistema más sencillo para cuando tenga que ser adaptado. Estos objetivos pueden relacionarse con los bloques elementales de los sistemas de información del modo siguiente:

- **Personas:**

En su mayor parte, la reingeniería es llevada a cabo por personal técnico y de sistemas de información.



- **Datos:**

Muchos proyectos de re-ingeniería son debidos a la necesidad de reestructurar los datos almacenados, ya sea para hacerlos más flexibles y fáciles de adaptar o para convertirlos a un nuevo entorno tecnológico.

- **Procesos:**

Muchos proyectos de re-ingeniería intentan reestructurar o reorganizar programas de aplicación para hacerlos más fáciles de mantener o convertirlos a un nuevo entorno tecnológico (por ejemplo, el lenguaje). Muchos otros modifican los métodos de entradas y salidas en los programas (por ejemplo, de modo batch a modo on-line, o de modo on-line a interfaces gráficas de usuario).

- **Redes:**

Algunos proyectos de aplicación buscan modificar las aplicaciones para adaptarlas a una nueva tecnología de redes.

- **Tecnología:**

En su mayoría, los proyectos de reingeniería se deben a cambios en la tecnología o a la necesidad de aprovechar mejor la tecnología existente.

Aún cuando el sistema esta creado bajo los requerimientos del usuario, no están descartadas posibles modificaciones al sistema, con las que puedan aumentar su capacidad o agregar algún requerimiento más, que no se haya contemplado en su momento.

Por este motivo se propone crear un control de cambios en el que se podría llevar una estadística de cada una de las modificaciones realizadas en el sistema en donde se indicará fecha, motivo de la modificación, tipo de modificación, solución propuesta y persona que la autorizó. Además servirá para poder llevar un histórico del sistema y de cada una de sus modificaciones por falla del sistema, limitaciones del diseño y/o cualquier aumento de necesidades.



IV.5.2 Liberación del Sistema

La liberación de un sistema son todas las actividades que hacen que un sistema de cómputo con todos sus componentes estén disponibles para su operación.

IV.5.2.1 Actividades de la liberación del sistema

El proceso general de liberación consiste en una serie de actividades interrelacionadas con transiciones entre ellas. Estas actividades pueden ocurrir del lado del constructor o del lado del cliente o de ambos. Debido a que cada sistema de software es un caso único, se precisan procesos y procedimientos con actividades bien delineadas y planeadas. En resumen la liberación del sistema debe ser interpretada como un paquete que tiene que ser aterrizado como un proceso general de acuerdo a características y requerimientos particulares del proyecto.

El disparador de esta serie de actividades es el documento de aceptación de usuario que generan las pruebas del sistema, en el cual se solicita que el sistema desarrollado y probado sea llevado al ambiente final de operación.

Al final del presente tema se muestra el formato de solicitud de Liberación del sistema integral de control de reportes para celulares.

En general se involucran las siguientes actividades:

- Plan de liberación
- Instalación
- Activación
- Desactivación
- Actualización
- Desinstalar
- Retirar de operación el sistema



Revisemos cada una de estas actividades en particular:

- Plan de liberación

La actividad de la liberación es el siguiente paso del proceso de desarrollo. Incluye todas las operaciones de preparación del sistema para ensamblarlo y llevarlo al lugar del cliente. Por tanto, se deben determinar los recursos requeridos para operarlo y recolectar información para preparar las siguientes actividades del proceso de liberación.

- Instalación

Es la colocación inicial del software en el ambiente de operación final. Actualmente esta actividad se soporta con herramientas especializadas. Las dos sub-actividades involucradas son Transferencia y Configuración. El primero corresponde a llevar los diferentes componentes al ambiente final, mientras que el otro se refiere a las áreas de configuración y ensamble para asegurar que el sistema este listo para ser usado.

- Activación

Es la tarea de encender o iniciar la operación del software. Para un sistema sencillo simplemente se ejecutara una serie de comandos, mientras que para otro muy complejo involucra hacer que todos los sistemas y componentes relacionados estén listos para operar. En liberaciones grandes, la copia productiva es instalada en el servidor de producción, pero en otros casos, el software liberado deberá instalarse en diversos ambientes, producción, preproducción para reproducción de errores y en algunos casos en los ambientes de recuperación de desastres.

- Desactivación

Consiste en apagar los sistemas que son substituidos, esta actividad puede ir desde simplemente apagar el sistema actual y en otros casos respaldarlo, activarlo en modo lectura.



- Actualización

La actualización es un proceso de instalar nuevas versiones de un sistema o software. Esto puede ser considerado como una instalación de menor complejidad si consideramos que la mayoría de componentes esta ya instalado, sin embargo no es trivial. Al igual que la instalación involucra la transferencia y la configuración.

- Mantenimiento adaptativo

La actividad de adaptación es también un proceso que modifica el software que previamente ha sido instalado. Difiere de la actualización porque las adaptaciones son derivadas de eventos que cambian el entorno de operación, mientras que las actualizaciones son derivadas de cambios en el software.

- Desinstalar

Desinstalar es remover un software o sistema que ya no será requerido. Esto también involucra reconfiguraciones de otros sistemas relacionados o dependientes para no impactarlos al remover dicho sistema.

- Retirar el sistema

Finalmente, cuando un software es marcado como obsoleto y no hay mas soporte del mismo este llega al final de su ciclo de vida y es retirado de la operación.



EMPRESA

ATENTA NOTA

Para:

De:

REF:

ASUNTO: Liberación del Sistema Integral de Control de Reportes para Celulares

Hago de su conocimiento que la prueba efectuada en laboratorio del 20 de febrero de 2008, sobre la versión 1.0 del Sistema Integral de Control de Reportes para Celulares, resultó satisfactoria con relación a los requerimientos funcionales y los casos de prueba procesados.

Por lo anterior, solicito su liberación al ambiente productivo, apegándose al contrato correspondiente.

Sin otro particular, reciba un cordial saludo.

Atentamente

México, D.F., a 25 de febrero de 2008.



CONTRATO

Entre:

Domicilio:

CIF:

A continuación denominado el CLIENTE

Y EMPRESA, con C.I.F. XXXXXXXX y domicilio social en:

Se conviene entre ambas partes las CLÁUSULAS que a continuación se mencionan:

- EMPRESA es una compañía dedicada al desarrollo de software que ha desarrollado una serie de servicios y sistemas en el campo de la informática.
- El CLIENTE esta interesado en contratar dichos servicios de programación para su aplicación en un sistema informático de su propiedad.
- Reconociéndose las partes capacidad suficiente para contratar y obligarse, conciertan el presente CONTRATO DE PROGRAMACIÓN Y DESARROLLO DE PROGRAMA DE UN SOFTWARE Y SU IMPLEMENTACIÓN, que libre y espontáneamente sujetan a las siguientes:

PRIMERA. OBJETO DEL CONTRATO

El CLIENTE contrata a EMPRESA para que realice la labor de programación, consistente en la elaboración del programa de software descrito en el ANEXO I.



SEGUNDA. UBICACIÓN

Las actividades descritas podrán desarrollarse en el local que LA EMPRESA utilizara para su actividad profesional, o mediante la utilización del sistema informático propiedad del **CLIENTE**. En este último caso, LA EMPRESA desplazara a un técnico de su organización al domicilio del **CLIENTE**, para realizar dicha tarea.

TERCERA. RELACIÓN

La relación entre las partes tiene exclusivamente carácter mercantil, no existiendo vínculo **laboral** alguno entre el **CLIENTE** y el **personal** de LA EMPRESA que eventualmente este prestando sus servicios en el domicilio social de aquel.

A tal efecto LA EMPRESA declara que dicho personal esta contratado de conformidad con la **Ley**.

CUARTA. PRECIO Y FORMA DE PAGO

El pago de los servicios de programación realizados por LA EMPRESA queda reflejado convenientemente en un documento adjunto al presente contrato que figura como ANEXO II.

QUINTA. PROGRAMA RESULTANTE

Una vez finalizado **el trabajo** de programación, LA EMPRESA entregara al cliente un respaldo con el código objeto del programa resultante y un ejemplar del **manual** de aplicación, conteniendo las **normas** de uso del programa. Así como una bitácora de entregas continuas de cada 20 días de presentación de avances con firma de conformidad del Cliente.



SEXTA. PROPIEDAD INTELECTUAL

- El CLIENTE reconoce los **derechos** de Propiedad Intelectual de LA EMPRESA sobre el resultado de su labor de programación.
- Dichos derechos protegen tanto el programa de ordenador que pueda resultar, como los **datos**, listados, **diagramas** y esquemas elaborados en la fase de análisis, el **manual** de aplicación, los restantes **datos** y **materiales** de apoyo, los símbolos de identificación, las contraseñas, los números de usuario y los símbolo de **seguridad**.
- El CLIENTE reconoce que el producto de la labor de programación de LA EMPRESA le es cedido por **la empresa** contratada exclusivamente para su uso y no para su reproducción, cesión, **venta**, alquiler o préstamo -como dato- y se compromete a no ceder su uso parcial o total de ninguna forma y a no transmitir ninguno de los derechos que tenga sobre él en virtud de este contrato, así como a no divulgarlo, publicarlo, ni ponerlo de ninguna otra manera a disposición de otras personas, salvo de los empleados de su empresa que tengan que utilizarlo.
- El CLIENTE será también responsable del incumplimiento de estas obligaciones por parte de sus empleados o de terceros que accedieran a el por negligencia del usuario.

SEPTIMA. CAMBIOS

El sistema cubre con los requisitos estipulados por el cliente, mismos que son entregados y probados en cada presentación de avance del proyecto. En cuanto el cliente acepte cada módulo presentado a la empresa ya no se hace responsable de cambios que se soliciten desfasados, estos cambios se podrán llevar a cabo con un costo adicional.



OCTAVA. EXTINCIÓN DEL CONTRATO

Este contrato se extinguirá por las causas generales establecidas en el Código Civil y en el Código de Comercio y en especial, por incumplimiento de las obligaciones dimanantes de este escrito.

NOVENA. ACTUALIZACION

En el caso de que alguna o algunas de las cláusulas del contrato pasen a ser inválidas, ilegales o inejecutables en virtud de alguna norma jurídica, se consideraran ineficaces en la medida que corresponda, pero en los demás, este contrato conservara su validez.

Las partes contratantes acuerdan sustituir la cláusula o cláusulas afectadas por otra u otras que tengan los efectos económicos más semejantes a los de las sustituidas.

DECIMA. DURACIÓN Y ALCANCE DE LA GARANTÍA

La garantía se establece por un plazo de 30 días naturales a partir de la fecha de la instalación del programa informático en el equipo del CLIENTE. Si el CLIENTE no ha notificado la existencia de defectos a LA EMPRESA durante el referido plazo, se considerara que esta conforme en todos los aspectos con el funcionamiento del programa, renunciando, a partir de entonces, a cualquier reclamación.

Y para que así conste, y en prueba de conformidad y aceptación al contenido de éste escrito, ambas partes lo firman por duplicado y a un solo efecto en la fecha y lugar indicados en el encabezamiento.



EMPRESA**EL USUARIO**

Por:

Por:

Cargo:

Cargo:

Firma:

Firma:

Fecha:

Fecha:

CLIENTE

(sello y firma)

EMPRESA

(sello y firma)

ANEXO I AL CONTRATO DE DESARROLLO DEL SOFTWARE DEL SISTEMA INTEGRAL DE CONTROL DE REPORTES PARA CELULARES

Requerimientos generales del sistema

Para el desarrollo óptimo de nuestro sistema, definiremos los requerimientos que nos permitirán cubrir todas las necesidades existentes:

- Se desea un sistema simple de manejar con la finalidad de no requerir de especialistas en su operación, si no que al contrario, cualquier persona con una mínima capacitación sea capaz de utilizarlo.
- Para facilitar su manejo debe de contar con interfaces gráficas amigables con el usuario.



- Al contar con interfaces gráficas es necesario que sea posible utilizarlo con el mouse, el teclado y el manejo de íconos.
- Se requiere que el sistema sea capaz de interactuar de alguna manera con los formatos o archivos con los que se proporciona información para la actualización de catálogos de nuevos modelos de equipos.
- Controlar el número de reportes que se soliciten.
- Obtener comprobantes de servicio para la empresa y el cliente.
- Agilizar el proceso de entrega-recepción del teléfono celular.
- Contar con un archivo histórico de fallas (bitácora) generadas en los equipos.
- Contar con una base de datos propia que permita llevar un histórico de cada equipo recibido, reparado y puesto otra vez en servicio, así como de los irreparables.
- Elaborar un diagnóstico de las marcas más confiables en el mercado.
- Reducir la cantidad de información requerida al momento de recibir el teléfono celular ya que esta deberá estar contenida en catálogos.
- Validación del usuario, así como niveles de acceso

Requerimientos particulares por área

Área Administrativa

El sistema requiere de un módulo que nos permita gestionar de una manera óptima y eficiente la información con la que se cuenta.

- Información del personal que labora en la empresa
- Información de los clientes
- Administración de los catálogos que contienen las fallas más frecuentes
- Especificar los parámetros respectivos de las órdenes de servicio vencidas.



- Alta, baja y actualización de usuarios del sistema
- Cambio de contraseña para poder acceder al sistema
- Respaldo de la base datos

Atención al cliente

El área de atención al cliente necesita de un sistema fiable para el levantamiento de órdenes de servicio, así como para la consulta del status de los equipos que ingresaron para revisión al laboratorio y si estos están en garantía o fuera de ella, pero que sobre todo este proceso sea de manera ágil al realizar la consulta y funciones como:

- Consulta de información referente al status del equipo.
- Verificación de tiempo de garantía.
- Captura mínima de datos en el levantamiento de órdenes de servicio.
- Generación de comprobante de orden de servicio.
- Reportes pendientes por atender

Laboratorio

El área de laboratorio requiere de un módulo del sistema en el que pueda ingresar un diagnóstico del equipo, una vez realizada una revisión técnica de la falla o posible falla, así como un presupuesto de la reparación si es que el equipo no entró en garantía.

- Orden de servicio
- Reporte general de fallas
- Folio de garantía
- Tiempo requerido para la reparación
- Catálogo con las fallas más frecuentes
- Nombre del técnico que realiza el servicio
- Cierre de la orden de servicio



Facturación y cobranza

Si la reparación del equipo se realizó mediante el respectivo cargo de la reparación por no entrar en garantía el área de facturación

- Notificar el costo de la reparación del teléfono
- Mostrar en pantalla toda la información de cliente, de ser necesario actualizar la información.
- Expedición de recibos de pago o facturas para la empresa y el cliente.
- Permitir la actualización de pagos, e identificar aquellas cuentas que aun han sido saldadas.

Área de Entrega

Una vez reparado el equipo se procede a la entrega del mismo al cliente, en este módulo se debe de contar con:

- Desplegar el número de la orden de servicio, el folio de la garantía, las especificaciones del tipo de teléfono, el tipo de falla y las observaciones pertinentes
- Hora de salida del equipo, número de recibo, nombre de la persona que recibe.
- Forma de pago (tarjeta de crédito o efectivo)

ANEXO II AL CONTRATO DE DESARROLLO DEL SOFTWARE DEL SISTEMA INTEGRAL DE CONTROL DE REPORTES PARA CELULARES

FORMA DE PAGO:

El precio final de la venta del sistema que figura en el ANEXO I, se establece en pesos., pagaderos de la siguiente forma:



- 30% en el momento de la firma del presente contrato.
- 50% en el momento de la instalación del programa.
- 20% a los treinta días de su instalación.

Correrán a cuenta del cliente los gastos que se ocasionen con motivo del servicio encargado.

EMPRESA

EL USUARIO

Por:

Por:

Cargo:

Cargo:

Firma:

Firma:

Fecha:

Fecha:

IV.5.3 Acuerdos de Niveles de Servicio

SLA son las siglas de la frase en inglés Service Level Agreement, que significa Acuerdos de Nivel de Servicio y a veces se abrevia como ANS. Un SLA es un protocolo plasmado normalmente en un documento de carácter legal por el que una compañía que presta un servicio a otra se compromete a prestar el mismo bajo unas determinadas condiciones y con unas prestaciones mínimas.

El nivel de servicio se basa en indicadores que permiten cuantificar de manera objetiva determinados aspectos del servicio prestado. Un indicador de nivel de servicio puede ser el tiempo de resolución de incidencias. Este indicador se mide a través de aplicaciones de gestión de incidencias que registran el momento que una incidencia es comunicada y cuándo es cerrada. La diferencia entre estos dos datos es el indicador en bruto desagregado que luego puede ser procesado mediante algoritmos para obtener promedios, desviaciones y otros indicadores normalizados.



En un SLA se pueden establecer tantos indicadores como se estime necesario y de su evaluación se obtienen por ejemplo penalizaciones a la empresa suministradora, identificación de puntos débiles del proceso e indicaciones para procesos de mejora continua en determinadas actividades.

El conjunto de indicadores de nivel de servicio suele formar un cuadro de mando donde se puede ver de manera global cómo se está desarrollando la prestación del servicio e identificar puntos críticos del proceso y establecer alarmas.

SLA también es una referencia a la hora de establecer parámetros de calidad del servicio basados en indicadores objetivos que obvian impresiones y percepciones más subjetivas y personales.

IV.6 Generación de Reportes.

Dado que es necesario llevar una relación de los reportes levantados dentro del sistema, es necesaria la realización de reportes de control para la empresa como para el cliente.

Reporte del proceso de Recepción o Entrada

Este reporte contiene los datos del cliente, la posible falla del dispositivo celular, el número de orden de servicio levantado, la cotización del trabajo, los accesorios con los que se cuenta al momento de recibir el dispositivo celular y algunas observaciones acerca del estado del mismo.

A continuación se representa en la figura IV.6.1 el modelo del reporte



ENTREGA, FECHA:

ORD.SERV.:

CLIENTE:

COTIZACION:

FALLA:

ACCESORIOS

GOMA		CARGADOR		BATERIA		ANTENA	
TAPA		HOSTLER					

MODELO	TRANSCTOR	MSN	ESN/SCN	IMEI	APC	TARJETA SIM	GARANTIA

OBSERVACIONES:

SI EL PRESUPUESTO NO ES ACEPTADO O EL TELEFONO ES IRREPARABLE, EL COSTO POR REVISION ES DE \$ 50.00

*** EN ACCESORIOS MOTOROLA NO APLICA LA GARANTIA ***

Toda reparación, (excepto equipos mojados y casos especiales), tiene una GARANTIA de 60 días en material y mano de obra, a partir de la fecha de entrega. La garantía por la reparación (con cargo al usuario) se hará efectiva unicamente presentando este documento; siempre y cuando el equipo presente idéntica falla o que la falla este relacionada directamente con aquella por la cual fué traído a ISATEL oroginalmente para su reparación.

Cumplido el tiempo de reparación se cuenta con 30 días naturales para recoger el equipo o de lo contrario la Empresa no se hace Responsable por el mismo.

Figura IV.6.1 Reporte Proceso de Entrada



Reporte de la Relación de Laboratorio.

Así mismo, dentro de la empresa es necesario llevar el control de los dispositivos que entran al laboratorio para alguna reparación o diagnóstico solicitado al equipo por los clientes.

Este reporte contendrá los siguientes datos: Orden de Servicio, Datos del Cliente o Empresa, Modelo del Dispositivo Celular, Posible falla del dispositivo, datos del equipo como lo son IMEI, ESN, Carrier y demás, así mismo, accesorios adicionales al dispositivo. Observaciones y si existe garantía del mismo, como lo muestra la figura IV.6.2.



FECHA DE ENTRADA AL LABORATORIO: _____ ORDEN SERVICIO: _____

CLIENTE/EMPRESA : _____ COTIZACION: _____

FALLA: _____ FECHA SALIDA: _____

EQUIPO CELULAR

MODELO: _____	IMEI: _____	ESN: _____	CARRIER: _____
---------------	-------------	------------	----------------

ACCESORIOS.

GOMA: <input type="checkbox"/>	CARGADOR: <input type="checkbox"/>	BATERIA <input type="checkbox"/>
ANTENA <input type="checkbox"/>	TAPA <input type="checkbox"/>	HOTLER <input type="checkbox"/>
TAPON <input type="checkbox"/>	TAPON <input type="checkbox"/>	TARJETA SIM <input type="checkbox"/>
MEMORIA <input type="checkbox"/>		

OBSERVACIONES: _____

*******EN ACCESORIOS NO APLICA LA GARANTIA**

Toda reparación,(excepto equipos mojados y caos especiales), tiene una GARANTIA de 60 días en material y mano de obra, a partir de la fecha de entrega

La garantía por la reparación (con cargo al usuario) se hará efectiva únicamente presentando documento, siempre cuando el equipo presente idéntica falla o que la falla este relacionada directamente con aquella por la cual fue traído a Laboratorio originalmente para su reparación.

Cumplido el tiempo de reparación se cuenta con 30 días naturales para recoger el equipo o de lo contrario la Empresa no se hace responsable por el mismo.

Figura IV.6.2 Reporte: Relación de Laboratorio



Por último es indispensable entregar un recibo de pago o recibo de entrega final al cliente con las características que se hayan realizado o modificado al dispositivo celular.

Este recibo contendrá lo siguiente: Fecha de entrega, orden de servicio, costo total, datos del cliente/empresa, la falla detectada especificada con detalle, accesorios reemplazados, observaciones de entrega, como lo muestra la figura IV.6.3



ENTREGA, FECHA:

NO. RECIBO:

CLIENTE:

ORD.SERV.:

COSTO:

FALLA:

ACCESORIOS

GOMA		CARGADOR		BATERIA		ANTENA	
TAPA		HOSTLER					

MODELO	TRANSCTOR	MSN	ESN/SCN	IMEI	APC	TARJETA SIM	GARANTIA

OBSERVACIONES:

OBS. ENTREGA:

RECIBI

*** EN ACCESORIOS MOTOROLA NO APLICA LA GARANTIA ***

Toda reparación, (excepto equipos mojados y casos especiales), tiene una GARANTIA de 60 días en material y mano de obra, a partir de la fecha de entrega. La garantía por la reparación (con cargo al usuario) se hará efectiva unicamente presentando este documento; siempre y cuando el equipo presente idéntica falla o que la falla este relacionada directamente con aquella por la cual fué traído a ISATEL oroginalmente para su reparación.

Cumplido el tiempo de reparación se cuenta con 30 días naturales para recoger el equipo o de lo contrario la Empresa no se hace Responsable por el mismo.

Figura IV.6.3 Reporte Relación de Laboratorio



Conclusiones



Se logró desarrollar un sistema hecho a la medida, cubriendo los aspectos deseados al aceptar la información técnica que permite recibir los celulares que sufren de alguna falla.

Se fundamentó la solución del sistema al contar con una base de datos relacional, que nos permite manejar la gran cantidad de información requerida; logrando un beneficio extra al contar con registros que permiten conocer que equipos son propensos a fallas y así evitar invertir en la adquisición de éstos, minimizando las pérdidas para la compañía debido a que se tiene una menor cantidad de equipos en garantía y generando ingresos por concepto de reparación.

El sistema diseñado cumple con el objetivo de tener una herramienta de apoyo para agilizar los procesos administrativos, de gestión y técnicos, de una empresa que brinda servicios de reparación de equipos de telefonía celular, apoyando las funciones del personal que atiende directamente a los clientes desde que es solicitado el servicio, hasta que se devuelve el equipo reparado, dando un seguimiento más detallado y mejores beneficios en la productividad y eficiencia laboral del personal de la compañía, así como un mejor servicio.

Gracias a los catálogos que se tienen almacenados en el sistema, los cuales son actualizados de manera continua, podemos registrar toda la información referente a los celulares que presentan una falla; ya sea por garantía o bien por costo de reparación, agilizando los tiempos de atención y solución.

Se implementó un ambiente visual de tal forma que el aprendizaje del manejo del sistema se facilitó por parte del personal, generando con esto un menor tiempo de capacitación y por consiguiente reducción en los gastos de la compañía.



Así mismo, éste sistema permite generar comprobantes de servicio para el usuario por parte de la empresa, brindando al cliente un respaldo en la reparación del equipo.

Los beneficios que se adoptaron repercuten en menores costos debido a la reducción del tiempo utilizado por los empleados, para capturar los datos de un nuevo celular o reingresar algún equipo anteriormente reparado, así como en la búsqueda de información que se necesite para llevar a cado alguna compostura; logrando minimizar el tiempo de registro del cliente, mejorando la calidad del servicio prestado.

Finalmente externamos nuestro agradecimiento al apoyo recibido por parte de la Facultad de Ingeniería y del Programa de Apoyo a la Titulación, por prepararnos para enfrentar los retos que la sociedad demanda.



Bibliografía



Introducción a los Sistemas de Bases de Datos C.S. Date
Addison Wesley
Iberoamericana.

Redes de Computadoras
Andrew S. Tanenbaum.
Prentice Hall.

Análisis y Diseño de Sistemas de Información
James A. Senn
Mc Graw-Hill.

Ingeniería de Software un Enfoque Práctico
Roger S. Pressuman
Ed. Mc Graw-Hill
México 2006 4ª edición.

Diseño y Gestión de Sistemas de Bases de Datos
Lucas Gomez Angel
Editorial Parainfo.

Aprende Microsoft SQL Server 2000 ya
Rebecca M. Riordan
Mc Graw-Hill.

Administración de bases de datos
Mark L. Gillenson
traducción Fernando Roberto Pérez Vázquez
México : Limusa-Wiley, 2006



La Biblia de SQL Server 2000
Patrick Dalton, Paul Whitehead
Madrid : Anaya Multimedia, 2001

Ingeniería de software orientado a objetos
Bernd Bruegge, Allen H. Dutoit
traducción Sergio Luis Maria Ruiz Faudon
México : Pearson Educacion, 2002

Fundamentos de Administración.
LOURDES MUNCH GALINDO
Trillas.

Introducción a los Sistemas de Base de Datos
C.J.DATE
Addison-Wesley Iberoamericana

Organización de las Bases de Datos
JAMES MARTIN
Prentice Hall

Fundamentos de Base de Datos
FORTH/SILBERSCHARTZ
McGraw Hill

Introducción a las Bases de Datos
MARK L. GILLENSON
McGraw Hill



Oracle Database 10g
Manual del Administrador
LONEY KEVIN
BRYLA BOB
Osborne,McGraw Hill

Oracle 10g
Administración y análisis de Bases de Datos
PEREZ LOPEZ CESAR
Alfaomega

Aprendiendo Visual Basic 6 en 21 días
PERRY GREG
Prentice Hall,Pearson

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/intfeatur esInSQLserver2000.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://www.desarrolloweb.com/articulos/262.php>
- http://es.wikipedia.org/wiki/Visual_Basic
- <http://www.isatid.net/mm/articulos/sql/sqldbt.htm>
- <http://www.elprisma.com/apuntes/curso.asp?id=6049>
- <http://www.monografias.com/trabajos4/costos/costos.shtml>
- <http://www.dow.com/wireenduser/sp/editions/april2005/total.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://www8.zdnet.com/pcmag/issues/1415/pcm00122.htm>
- http://es.wikipedia.org/wiki/Diagrama_de_bloques_de_modelo_matem%C3%A1tico
- <http://www.esi2.us.es/~vargas/docencia/tca/ejercicios/ejerciciosCont2.pdf>



Manual de Usuario



MANUAL DE USUARIO

El presente manual pretende dar a conocer a los operadores las características y las formas de funcionamiento del software.

Abra a la aplicación de la base de datos en donde se le preguntará el usuario y contraseña de acceso, figura1.



Figura 1

Después de ingresar su usuario y contraseña, presione el botón enter. Se mostrara la figura 2, donde encontrará las opciones principales del sistema:



Figura 2



La opción de administración desplegará las opciones de personal, usuarios, catálogos, parámetros y carga de archivos en Excel (figura 3).



Figura 3

En la opción de personal se agrega al personal a cargo del Centro de Atención a Clientes, donde se proporcionan los datos de identificación así como la asignación de un ID único que lo identifica dentro de la empresa (figura 4).



Figura 4



En la opción de parámetros, se permite asignar el número de días en el cual se considerará que una orden de servicio ha vencido (figura 7).



Figura 7

Finalmente la opción de cargar archivo de Excel, permite abrir los archivos de Excel que se hayan registrado en el sistema (figura 8).

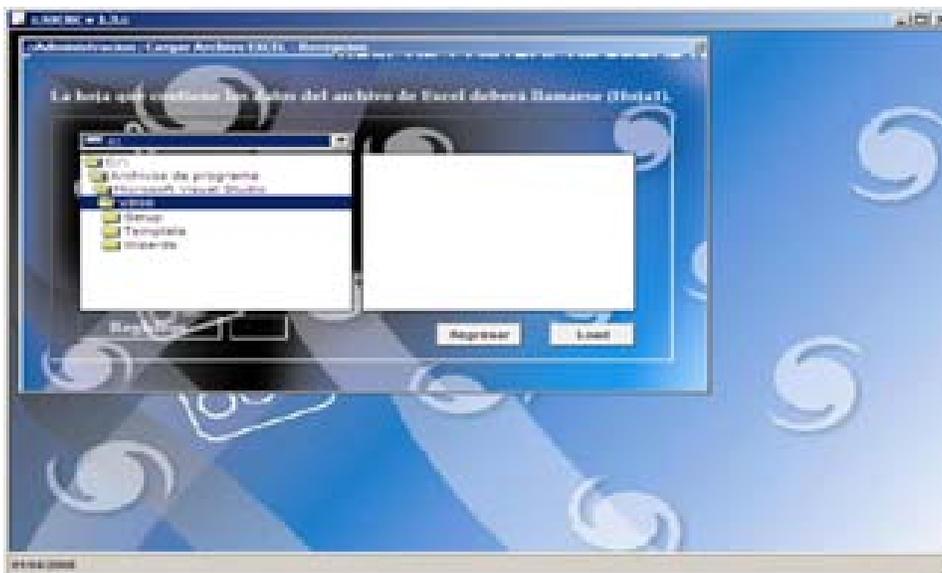


Figura 8



En el menú cliente, se despliegan las opciones de alta y consultas (figura 9).



Figura 9

Si damos clic en alta se desplegará una pantalla donde se preguntan los datos del nuevo cliente (figura10).



Figura 10



La opción de consulta permite realizar búsqueda de cliente ya registrados en la base de datos, en este caso solo se deberá teclear el ID del cliente para su búsqueda o ingresar el nombre del usuario o compañía según sea el caso (figura 11).



Figura 11

En el menú recepción se despliegan las opciones de captura, cambios y consultas (figura12).

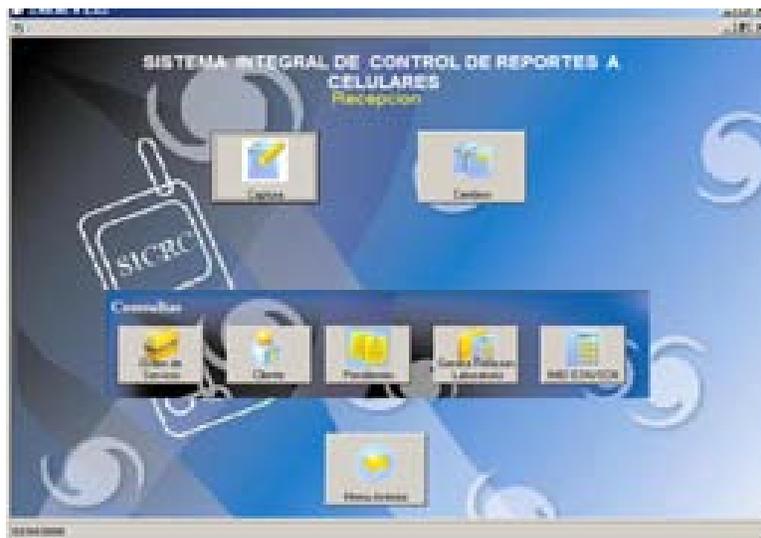


Figura 12



Para la opción de consultas se pueden realizar búsquedas por orden de servicio, clientes, pendientes, relación de laboratorio o IMEI (figura15).



Figura 15

En el menú de laboratorio se dan las opciones de captura y consultas (figura16).

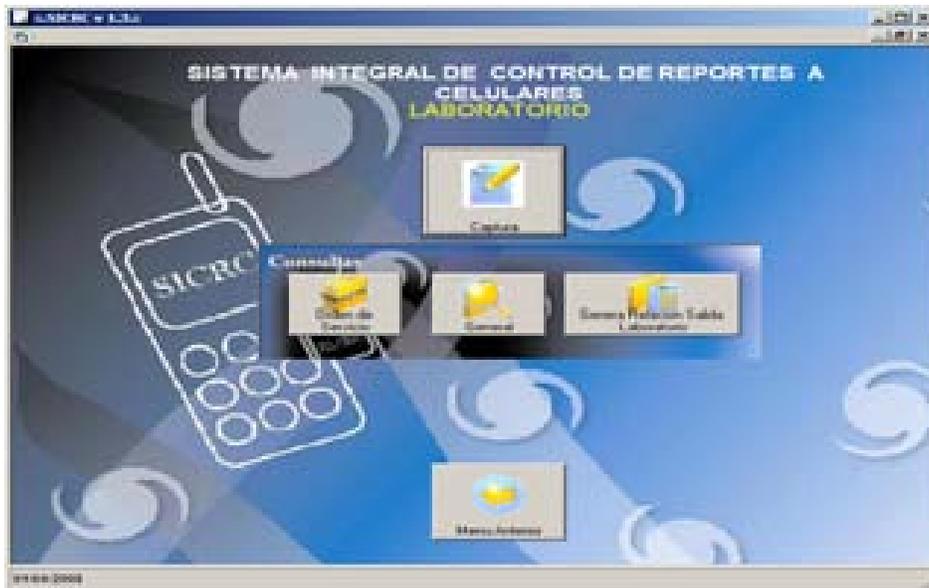


Figura 16



Si seleccionamos la opción de captura se deberá de incluir la información referente a la reparación del equipo celular (figura 17).

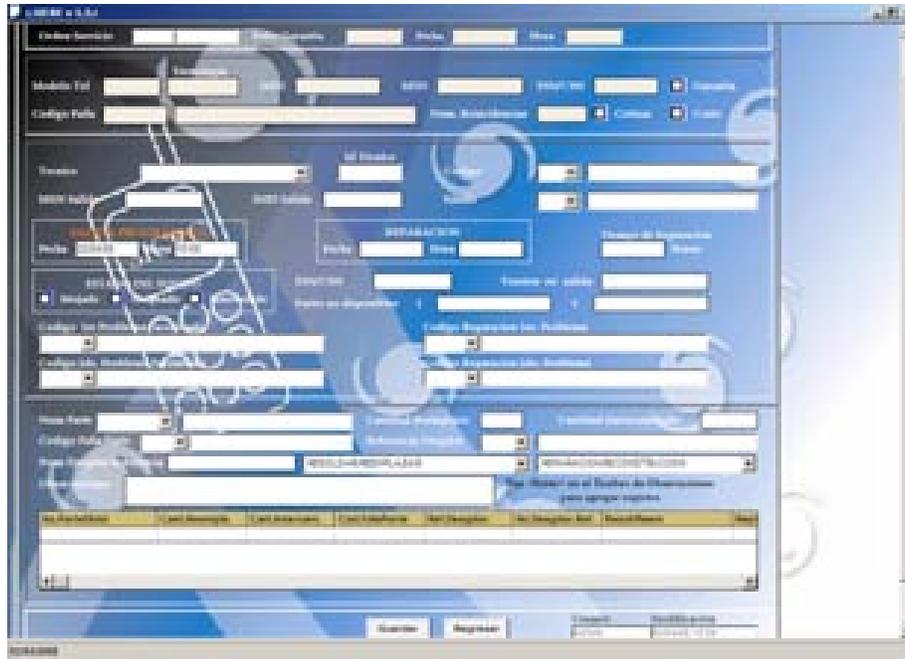


Figura 17

Si se selecciona la opción de consultas, podremos realizar estas por medio de orden de servicio, de manera general o por salida del laboratorio (figura18).

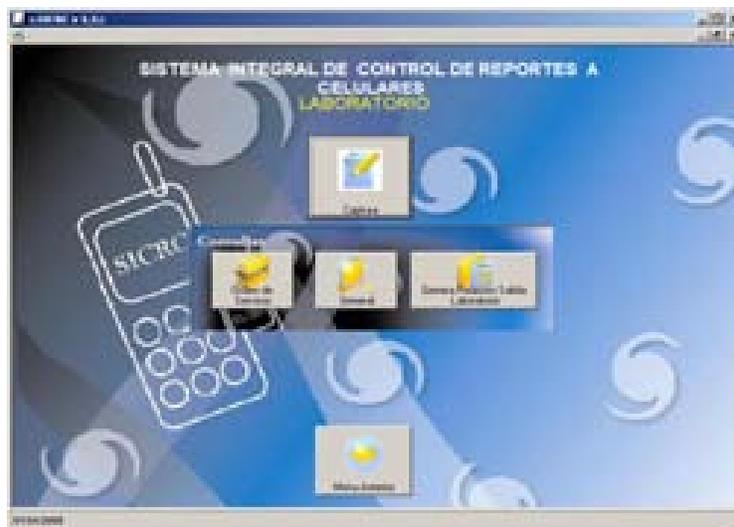


Figura 18



En caso de seleccionar la opción de consultas, estas se podrán realizar por orden de servicio o pendientes (figura 21).



Figura 21

En el menú recibos se podrá realizar el abono o el pago de una orden de servicio (figura22).



Figura 22



En el menú de consultas generales, se podrán realizar por pendientes de atención, atendidas por empleado o IMEI del equipo (figura 23).



Figura 23

La opción del menú de cambiar password permite el cambio de contraseña para poder acceder a la base de datos (figura 24).



Figura 24



Finalmente la opción de salir en el menú cerrara la aplicación, (ver figura 25).



Figura 25