

UNIVERSIDAD NACIONAL DE INGENIERIA

Facultad de Ingeniería Industrial y de Sistemas



**INSTALACION DE UNA RED DE COMPUTADORES
EN UNA EMPRESA DE TRANSPORTES MEDICOS**

Informe de Ingeniería

**Para Optar el Título Profesional de:
INGENIERO DE SISTEMAS**

JUAN MANUEL CHAVEZ HORNA

LIMA-PERU

1993

Indice

	Pág.
1.- Descripción de la empresa	
1,1 Definición de la empresa	1
1.2 Forma manual de Trabajo	1
1.3 Objetivos a alcanzar	2
2.- Justificación de un sistema de red de computadores	
2.1 Por reducción de costos	3
2.3 Por productividad	3
3.- Evaluación de sistemas computarizados	
3.1 Red Novell	6
3.2 Red Banyan	7
3.3 Red Lantastic	7
3.4 Sistema IBM/AS400	8
3.5 Elección del sistema	9
4.- Instalación de la Red	
4.1 Hardware	12
4.2 Software	13

5.- Desarrollo del Software

5.1 Consideraciones Generales	16
5.2 Requerimientos específicos	17
5.3 Ambiente de desarrollo	18
5.4 Pantallas de la aplicación	20

6.- Recursos Humanos

6.1. Para la instalación de la red	37
6.2 Personal operativo	37
6.3 Mantenimiento de la aplicación	38

7.- Puesta en marcha de la aplicación

7.1 Trabajo en paralelo	39
7.2 Orden de automatización	40

8.- Conclusiones

8.1 Conclusiones Generales	41
8.2 Respecto a la aplicación	41

9.- Anexos

9.1	Listados de los programas.....	43
9.2	Facturas emitidas por la aplicacion	93
9.3	Tarjeta del paciente	94
9.4	Requerimientos de pagos	95
9.5	Beneficio - Costo	96
9.6	Bibliografia	97

1.- Descripción de la empresa

1.1 Definición de la empresa

La empresa es una compañía de ambulancias llamada '**Transportes Medicos**', (Tambien opera bajo el nombre de '**American Medical, non-emergency transportation**') ofrecen el servicio de transporte de pacientes para casos de no emergencia (no involucran riesgo de atención inmediata). La empresa se encuentra localizada en la ciudad de Miami, USA, en el condado de Dade. Transportes Medicos realiza el transporte de personas inválidas, ancianos, transporte en camilla, pacientes con fracturas menores, etc. generalmente desde la casa de los pacientes hasta las clínicas u hospitales y viceversa.

El transporte que se ofrece es al paciente, sin embargo son los hospitales, clinicas ó compañías aseguradoras las que solicitan los servicio de '**Transportes Medicos**'. Por lo tanto el pago del servicio lo realizan los hospitales ó las empresas de seguros. Transportes Medicos realiza la facturación tan pronto como se culmina el servicio y envia las facturas a sus diferentes clientes (hospitales y/ó aseguradoras).

1.2 Forma manual de Trabajo

Antes de tener un sistema computarizado, '**Transportes médicos**' contaba con 3 computadores personales, no interconectados, utilizando paquetes comerciales contables standard (Lotus 123 para cuadros, Wordperfect para trabajos de oficina, ACCPAC de Computer Associates para la Contabilidad) para el manejo de sus operaciones. 8 Ambulancias prestaban servicio a un aproximado de 35 clientes (en este caso 'clientes' son los hospitales, o aseguradoras) que en promedio diario transportaban alrededor de 35 a 40 pacientes al dia. La empresa contaba con 45 empleados (no incluye personal temporal).

Cuando una empresa (generalmente un hospital, aseguradora ó un consultorio medico) solicitaba los servicios de '**Transportes medicos**' para transportar a algun paciente, los operadores tomaban la orden y registraban los datos del paciente en formatos pre-establecidos. Luego de acuerdo a la hora y fecha del transporte se programaba un rol de transportes, todo este proceso era manual y no involucraba un computador.

Luego las ambulancias realizaban el recojo de los pacientes, al finalizar con un paciente, el chofer de la ambulancia, la cual estaba equipada con equipos de radio, llamaba a Transportes medicos para informar la distancia recorrida, los operadores de radio en la oficina de la empresa, registraban manualmente la distancia y enviaban esta información al departamento de facturación.

El departamento de facturación calculaba, también manualmente el valor a facturar basado en la distancia, tipo de servicio (existen tres tipos de servicio: camilla, silla de ruedas y ambulatorio, los servicios eran regulados por el departamento de salubridad del condado de la Florida), y basado en una tarifa (la cual se elabora tomando en cuenta el tipo de empresa y generalmente el volumen de servicio que prestaba a la empresa aseguradora u hospital). La factura una vez calculada era ingresada en computadores personales no interconectados, los cuales generaban las facturas correspondientes.

La información que generaban los computadores personales: como son impresión de facturas, eran enviados por correo a los diferentes clientes, otros reportes como cuentas por cobrar, totales de facturas, cheques, etc. era utilizada por una empresa externa de contadores que manejaba la contabilidad para 'Transportes medicos'.

1.3 Objetivos a alcanzar

'Transportes Medicos' tenía demasiados problemas manejando esta información manualmente, su objetivo era automatizar en la medida de lo posible y en forma económica, el proceso de tomar ordenes hasta realizar la facturación en forma automática.

Asi mismo habia decidido realizar una significativa expansión, asi que era primordial contar con un sistema mas efectivo de control y mas rápido, debido a que el proceso era manual, muchas veces la información no se encontraba a tiempo y por lo tanto el paciente tenia que ser referido a otra empresa o se cancelaba ese transporte, lo cual significaba pérdida significativa de ingresos, ya que se había llegado a un punto en que las cancelaciones eran demasiadas.

2.- Justificación de un Sistema Computarizado

2.1 Por reducción de costos

'Transportes médicos' había planeado realizar una inversión significativa en su flota de ambulancias, de 8 unidades había decidido triplicar su capacidad de transporte a 25 unidades (mediante adquisiciones de pequeñas empresas), así mismo realizar una fuerte inversión en publicidad, lo cual significaba poder manejar la nueva demanda de servicios que iba a encarar. Se había calculado un incremento de 180 nuevos hospitales y aseguradoras que iban a solicitar los servicios de 'Transportes Médicos', así mismo el transporte se había estimado en 80 a 120 pacientes por día, lo cual significa alrededor de 480 a 720 pacientes a la semana.

El método de trabajo que se tenía antes de implantar el sistema computarizado era manual, y aun así, era bastante engorroso el poder controlar cada uno de los pacientes que se tenía que transportar, ya que se tiene que guardar un registro del viaje por cada paciente, el tipo específico de servicio (el precio era diferente si en el transporte se utilizaba un camilla, equipo especial de oxígeno, silla de ruedas, etc.), el hospital o aseguradora que solicitaba el servicio, la distancia recorrida, datos clínicos del paciente, etc.

Antes de la implantación del sistema computarizado, existían 9 empleados dedicados exclusivamente al registro y recolección de la información. Con el sistema manual se pensaba contratar 8 nuevos empleados. Se había calculado que era más costoso contratar nuevos empleados que adquirir un sistema computarizado y reentrenar a los empleados existentes.

2.3 Por productividad

Si se implantaba un sistema de red de computadores, se podía, adicionalmente beneficiar de la utilización de paquetes de software, para la empresa en general, por ejemplo los documentos internos de la oficina eran manejados con un procesador de textos, y la mayoría de empleados habían aprendido a utilizarlos.

Por ejemplo Lotus es un paquete muy popular entre los empleados de Transportes Médicos, al utilizar este paquete para realizar cuadros, los otros empleados tenían acceso casi inmediato a los resultados que se pudiera tener de estos cuadros. Lotus era utilizado en la empresa para realizar cuadros estadísticos y resúmenes de las actividades de la empresa. Así mismo se veía un beneficio en utilizar un procesador de textos en versión 'server', ya que un posible documento podía ser utilizado por otros empleados, siempre y cuando tuvieran el correspondiente acceso.

Otra posibilidad que otorga una red es el correo electrónico que agiliza la comunicación entre las diferentes oficinas de la empresa, como son memorandums, cartas, reuniones, etc..últimamente se esta dando una mayor importancia al correo electrónico ya que reduce en gran medida la cantidad de papeles que se tiene en la oficina. En el mercado de redes estan apareciendo productos que permiten la intercomunicación entre diferentes departamentos de la empresa que tienen acceso a diferentes computadores personales.

Productos como E-Mail (Microsoft, Inc.) , CC-Mail (Lotus Development, Inc.) tienen gran popularidad para la transmisión de documentos y correo electrónico en empresas de pequeña y mediana envergadura. Cada vez esta tomando mayor importancia el tipo de software de correo electrónico que se va a utilizar antes de decidir por la compra de hardware.

Otro punto importante en la compra de la red era que se podía enviar electrónicamente las facturas a algunos hospitales y empresas de seguros que contaban con computadores propios y tenían la capacidad de recibir información via 'modem' , lo que reducía significativamente el tiempo en cobrar las facturas de semanas a días. En la compra de los equipos se debía considerar, la interface de comunicación con estos computadores. La mas importantes clientes de Transportes medicos trabajaban con redes Novell, por lo tanto la comunicación con este tipo de redes debe y era considerada como un fuerte punto en la decisión de compra de nuevos equipos.

3.- Evaluación de Sistemas Computarizados

Una vez establecida la necesidad y justificación de un sistema automatizado de datos, era necesario definir cual es el más conveniente para la empresa.

Tomando en cuenta varios factores como son el tamaño de la empresa, inversión a realizar, comunicación con los otros sistemas computarizados de los principales clientes de Transportes Medicos (para realizar la transmisión electrónica), se definieron los siguientes sistemas computarizados:

- Red Novell
- Red Banyan
- Sistema IBM/AS400
- Red Lantastic

No se contemplaron sistemas mas grandes tipo 'Mainframe', debido a que el movimiento de información que se tenía y el flujo de información proyectado no justificaba una inversión tan grande; asi como también un presupuesto de gasto que no podia ser excedido. Por otro lado la tecnología de procesamiento 'Mainframe' (Sistemas IBM 4341, 4381, 3900, etc.), estan siendo desplazadas por tecnologías más eficientes y mas baratas de adquirir (Por ejemplo redes de microcomputadores, Procesamiento Cliente/Servidor, Programacion por Objetos, Software Grafico, etc.). Asi mismo por solicitud del cliente no se consideraron sistemas basados en UNIX ó XENIX y en PATH WORKS por considerar una tecnologia sofisticada que requiere de intenso entrenamiento y mayor inversion de la que se habia planeado.

A continuación se muestra un breve análisis de las diferentes redes consideradas como una solución para la empresa. Debe notarse que la fecha en que se realiza este estudio es entre Diciembre de 1992 y Enero de 1993, debido a la rapidez con que evoluciona la tecnologia tanto en software como en hardware, no se toman en cuenta productos que en aquel entonces no estaban comercialmente disponibles, por ejemplo se hace el anaálisis con Novell 3.11 en lugar de la ultima version de Novell 4.00, la cual aún no estaba disponible al momento de evaluar las redes.

Tampoco se considera en este estudio el sistema operativo Windows NT de Microsoft, que, aunque era muy prometedor para el tipo de sistema que se pensaba adquirir, no se encontraba comercialmente disponible.

3.1 Red Novell

Esta es la red mas popular actualmente en el mercado, desarrollada por la empresa Novell, Inc. establecida en Ohio, USA. Novell fué una de las primeras redes de computadores que aparecieron en el mercado, lo cual hizo que se posicionara como un fuerte suplidor de software para redes en el mundo entero, tiene más de 10 años en el mercado de computadores y es una empresa muy sólida, esto es muy importante de considerar, ya que las redes de microcomputadores son de tecnología reciente, hace mas 15 años esta tecnología ni siquiera existia, por lo tanto al escoger un software de este tipo, es muy importante ver cual es el soporte técnico que se tiene detras del producto.

Novell tiene casi un 50% del mercado de redes, es muy popular, y todo hace indicar que lo será por muchos años. Netware, el nombre comercial del software, consiste en la interconexión de computadores personales compatibles. También se pueden conectar computadores Apple Maccintosh, y practicamente con cualquier sistema de IBM, desde computadores personales hasta los sistemas IBM 3900

Esta red tiene la ventaja de poder utilizar paquetes del ambiente DOS (utilizados en las computadoras de 'Transportes Medicos',). Existen muchos paquetes de software para este tipo de red, por Ejemplo Lotus, Dbase, Procesadores de Texto tipo Wordperfect, Word. etc.

Novell ó Netware es el sistema de red que mas sub-productos tiene, es decir, los fabricantes de partes de computadores hacen sus productos compatibles con este tipo de red. Por ejemplo se pueden encontrar tarjetas de red de mas 10 fabricantes, software de comunicación, Fax-servers, conexión con innumerables tipos de impresoras, practicamente cualquier computador compatible puede ser conectado a una red Novell, sea una XT, 286, 386, 486.

Novell acepta diferentes tipos de topología (estrella, token-ring, bus, etc.) ademas la velocidad de transmisión es bastante aceptable (Ethernet de 10 megabits por segundo).

Una red Novell consiste de un computador principal (puede ser un 386, 486 o modelos sucesivos) llamado 'Server' ó la 'maquina servidora'; y varias estaciones conectadas al 'Server'. Las estaciones que se conectan a la maquina servidora pueden ser cualquier tipo de computador compatible, por supuesto, es necesario agregarle la correspondiente tarjeta de conexion al computador.

Cuando un computador personal se conecta a la maquina servidora, esta tiene acceso a todo ó parte de la informacion que se encuentra en el computador principal, segun los accesos que hayan sido definidos por el administrador de la red. Por ejemplo una estacion puede tener acceso al subdirectorio de Lotus y por lo tanto a utilizar la aplicacion, mientras que otra estación no tenga acceso a este paquete.

3.2 Red Vines ó Banyan

Desarrollada por la empresa Banyan, establecida en New York, USA. Esta red al igual que la anterior consiste de una 'maquina principal' ó 'Server' y varias computadoras compatibles conectadas al 'Server' en diferentes topologías (Estrella, bus, etc.).

El software de red Banyan es compatible con programas DOS, utilizando por supuesto la version de red. Sin embargo este tipo de red esta especificamente diseñado para redes conocidas como WAN (Wide Area Network) es decir para interconectar grandes redes, sean estas remotas ó locales. La empresa es sólida, es decir, no se corre el riesgo de adquirir un producto, que despues de algunos años no puede darsele servicio porque la empresa ya no se encuentra en el mercado. Aunque el soporte técnico de esta empresa es muy especializado y eficiente, los precios por el servicio eran los mas elevados que en los otros sistemas considerados.

Debe notarse que este tipo de red es mas sofisticado que la red Novell lo cual hace que una red Banyan otorgue mayor versatilidad de uso pero a un costo mas alto que la red Novell.

Redes Banyan mayormente estan siendo utilizadas cuando la empresa tiene varias 'sucursales' y en cada una de estas oficinas es necesario a sus vez tener una red de computadores. Logicamente el costo de una red Banyan es mucho mas alto que el de una red Novell. Ademas no existen mucha variedad de proveedores de partes para una red Banyan como lo hay para una red Novell.

3.3 Red Lantastic

Lantastic (de la empresa Artisoft, Inc.) es una red relativamente nueva (no mas de 5 años en el mercado), que se ha impuesto en el mercado debido a su bajo costo, sin embargo la performance que entrega una red de este tipo es muy baja (2 Megabytes por seg. comparada con Ethernet de Novell ó Banyan de 10 Megabytes por seg.).

Lantastic es una red que es muy aceptada para pequeñas redes de 4 a 6 estaciones. La forma de trabajo es de 'peer to peer', ó de punto a punto, esto significa que se pueden tener varios 'servers' interconectados, es decir en la red se pueden tener varias máquinas principales y diferentes estaciones interconectados hacia los 'servers'. Novell y Banyan tambien aceptan interconexion de servers pero bajo condiciones especiales que implican un mayor costo.

Aunque en teoria en una red Lantastic se pueden tener conectados hasta 300 estaciones, en la realidad se ha comprobado que el procesamiento en las estaciones se degrada en forma sustancial cuando se tienen varias estaciones accedando las

maquinas 'principales'. Otra desventaja de Lantastic es que la tecnología es 'propietaria' es decir no existen otros fabricantes de este tipo de red, esto significa depender de un solo suplidor. La ventaja mas importante probablemente es su bajo precio. Sin embargo la velocidad de operacion que es requerida la pone en desventaja frente a las otras redes.

En cuanto al soporte técnico que esta empresa otorga, es la más económica, por ejemplo, los técnicos de la empresa otorgan ilimitado soporte por teléfono sin costo adicional para el cliente. Existen consultores locales en la ciudad de Miami que tiene conocimientos especificos y detallados sobre esta red y cuyos costos de mantenimientos son los mas bajos comparados con las otras redes consideradas.

3.4 Sistema IBM/AS400

IBM (International Bussines Machines) es la empresa de computadores mas grande del mundo y que tiene su propio sistema a ser considerado. El computador AS400 tiene una tecnologia de avanzada y su velocidad de procesamiento es la adecuada y requerida para el tipo de trabajo en 'Transportes Medicos', Sin embargo el costo de utilizar estos equipos es muy alto.

Estos sistemas utilizan sus propias estaciones de trabajo, esto significa que si se desea utilizar los computadores personales existentes, se tiene que comprar equipo adicional costoso. Por otra parte el Sistema AS400 utiliza su propio sistema operativo, esto hace que no se tenga una buena variedad de paquetes comerciales (se tiene que comprar paquetes de software especiales y luego mandarlos a adaptar, ó diseñarlos completamente por empresas de software especializadas).

Otra desventaja es que si bien la tecnología es de avanzada, es 'propietaria', es decir solamente IBM fabrica estos sistemas. El adquirir un sistema AS400 significa estar atado a un solo abastecedor lo cual lo pone en desventaja al comprador, especialmente cuando puede encontrar en el mercado alternativas mas atractivas en cuanto a costo, servicio, partes, software, profesionales y personal tecnico, etc.

3.5 Elección del sistema

Basado en las consideraciones anteriores y el tipo de uso que se tenía en mente se estableció un sistema de red adecuado para la empresa, no se tomó en cuenta por ejemplo computadores XT, 286, ó 386SX por considerarse muy lentos para la aplicación que se tenía en mente.

El equipo de red comprende 20 estaciones y una maquina 'servidora' ó server 486 en una red Ethernet, cable coaxial RG58, tipo Bus, con las siguientes características:

Maquina servidora:

- Procesador 486 DX de 40 Mhz ó mas
- 16 Megabytes de memoria RAM, ampliable hasta 48 Megabytes
- 500 Megabytes de espacio en disco duro, ampliable hasta 2.5 Gigabytes, disco tipo SCSI de alta velocidad (acceso mínimo 10 milisegundos).
- Tarjeta de red Ethernet
- Monitor VGA monocromático
- 2 Disqueteras 3 1/2" y 5 1/4" de alta densidad.
- Al menos 6 'slots' disponibles para futura expansion.
- Equipo de 'Backup' es decir para realizar copias de seguridad de los datos
- UPS 1200 Watts, es decir baterías de respaldo, en caso de corte súbito del fluido eléctrico, con un mínimo de 45 minutos de duración e interface con la red.

Estaciones de Trabajo

- Procesador 386 DX, de al menos 25 Mhz ó mas de velocidad.
- 2 Megabytes de memoria RAM, ampliables hasta 16 Mb.
- 1 disquetera de 3 1/2"
- Monitor VGA color.
- Tarjeta de red NE-2000, cable RG58.
- Al menos 4 slots disponibles para futura expansion.

En caso de IBM se eligió el Sistema AS400 de IBM, con 5 estaciones no inteligentes (es decir no son computadores personales) y 10 computadores PS/2 conectadas al computador principal.

El cuadro siguiente esquematiza los gastos que se tendrían que efectuar en caso de elegir cada uno de los diferentes sistemas considerados. Se ha separado el costo de hardware y el costo de software, en cuanto a software, solamente se considera aquellos paquetes necesarios para poner en funcionamiento el sistema, es decir el sistema operativo, librerías y utilitarios que generalmente vienen con el paquete principal, no se considera el costo de desarrollar una aplicación.

Tipo de Red	Costo Hardware U.S. \$	Costo Software U.S. \$	Total U.S. \$
Novell 3.11	34,000	3,500	37,500
Banyan	38,000	6,500	44,500
AS/400	75,000	7,000	82,000
Lantastic	30,000	2,500	32,500

Costo de implantar una red

En cuanto a performance, desde que la red principalmente se dedicaba para aplicaciones comerciales y de la cual lo más importante era el poder trabajar con una Base de datos, es decir tener la capacidad de almacenar y procesar registros en todo en grandes cantidades, era importante saber cuál era el rendimiento en esta área.

No era primordial por ejemplo la velocidad de proceso del CPU como lo era la velocidad de acceso al disco duro. La red no iba a ser utilizada para desarrollo de aplicaciones sino más bien para 'correr' aplicaciones que iban a ser desarrolladas en otras redes.

La red a considerar debía ser capaz de almacenar por lo menos 400 Megabytes de información en línea en todo momento, ya que se había estimado el poder tener información sobre pacientes, traslados, facturación, etc. de por lo menos 24 meses pasados. Así mismo la velocidad de búsqueda en la red era más importante que, por ejemplo, la cantidad de memoria que iban a tener las estaciones.

Existen comparaciones sobre la velocidad de procesamiento de diferentes redes. en el siguiente cuadro se muestra la performance entregada por una red de 12 estaciones.

Este cuadro fue obtenido de las comparaciones técnicas que hace "Corporate Computing" (publicación de Noviembre de '92, Ziff Publishing company) la cual es una publicación técnica que analiza redes de diferentes marcas y topologías. Se tiene el siguiente cuadro basado en la lectura de una base de datos con 50,000 registros:

En este caso la base de datos es clasificada por un código, luego 2 campos de la base de datos es sumada y puesta en un tercer campo de la misma base. Aunque en cada plataforma se utiliza diferente producto para manejar la base de datos; de todas maneras es interesante realizar la comparación ya que nos permite apreciar las diferencias entre una plataforma y otra.

Tipo de Red	10 Estaciones	15 Estaciones	20 estaciones	25 estaciones
Novell 3.11	20 seg.	24 seg.	35 seg.	47 seg.
Banyan	17 seg.	19 seg.	25 seg.	38 seg.
AS/400	18 seg.	19 seg.	28 seg.	33 seg.
Lantastic	300 seg.	450 seg.	900 seg.	1400 seg.

Performance de redes al procesar una Base de datos.

En realidad las tres primeras redes se encuentran bastante alejadas de la red Lantastic en cuanto a performance. Sin embargo la Red Lantastic es la más barata, pero también la más simple y poco sofisticada.

Tomando en cuenta estas consideraciones se optó por instalar la red Novell, que es la 2da en cuanto a precio, pero que sin embargo muestra una consistente performance entre el resto de redes consideradas.

4.-Instalacion de la red

4.1 Hardware

La red Novell a instalar constaba del siguiente equipo:

- 15 Microcomputadores personales compatibles, del tipo 386 DX, con 2 Megabytes de memoria RAM, 2 puertos seriales, 1 puerto paralelo, disquetera de 1.2 Mb y 1.44 Mb, tarjeta de red Ethernet NE-2000
- 1 Server 486 DX de 66 Megahertz, 16 Megabytes de memoria RAM, 2 puertos seriales, 3 puertos paralelos, 600 Megabytes de capacidad
- Topologia Ethernet, Bus (thin Ethernet), cable RG58, 50 ohmios.
- Software Novell version 3.11 para 50 usuarios, server dedicado.
- 2 impresoras de alta velocidad conectadas a la red (600 cps). Estas impresoras estan conectadas al server, hacia las cuales se envian la facturacion y los tickets de impresion.
- 4 impresoras de baja velocidad para control de las estaciones de trabajo, reportes de uso local, mantenimiento de la red, etc.
- 4 modems de alta velocidad (9600 bps) conectada a las diferentes estaciones.

Se decidió conectar a la red los computadores que se tenían en la empresa, de esta manera se podían compartir los archivos en los que se estaba trabajando: Lotus y documentos internos en el procesador de textos. En el gráfico nro. 1 se muestra el esquema de la red a instalar.

La red desde el inicio debía soportar comunicación con otros equipos de red, ya que algunos clientes contaban con sistemas computarizados y la comunicación se hacía a través de modems, se instalaron modems de alta velocidad (9600 bps) en cuatro estaciones.

4.2 Software

Para la red se habia pensado desarrollar una aplicacion especial, la cual iba a constituir la razón de la utilización del sistema computarizado; adicionalmente se habia decidido adquirir paquetes comerciales. Todos estos paquetes son para el ambiente DOS (Disk Operating System). Sin embargo se habia cargado Microsoft Windows en algunas máquinas, El paquete a desarrollar que va servir como base de uso del sistema se explica en un capítulo aparte, sin embargo para los paquetes comerciales se habia pensados instalar:

Lotus 123.- 'Transportes medicos' ya contaba con personal que conocía y utilizaba este paquete de uso popular por lo tanto se habia pensado instalar la version de red de este paquete, este paquete se había destinado para elaborar cuadros estadisticos sobre el movimiento de pacientes entre los diferentes centros hospitalarios.

Wordperfect.- Asi mismo todos los documentos que se elaboraban en la empresa era con la utilizacion de este paquete, igualmente se habia pensado instalar la version de red de este popular programa. Este paquete era basicamente para elaborar documentos en la empresa, aunque el paquete a instalar era para ambiente DOS, se había adquirido también la version para Windows, la cual permite una mayor calidad de impresión.

Para las comunicaciones se eligio 'PC anywhere' en version de red. Lo que permite este paquete es transferir y extraer informacion de computadores y/ó redes remotas. Las estaciones y el server se cargaron con DOS 5.0, es importante estandarizar el sistema operativo en una red para evitar problemas de versiones.

Foxpro.- Este paquete es para desarrollo de aplicaciones, aunque no se habia pensado instalar para desarrollar alguna aplicacion internamente, la empresa consultora que se encargaria de desarrollar el paquete habia instalado este paquete para control cuando la aplicacion estuviera operando.

El correo electrónico fué uno de los paquetes que se tuvo en mayor consideración al analizar la compra del hardware, en el caso de Novell, se tienen varias alternativas, es decir existen mas de una docena de paquetes para realizar correo electrónico: Microsoft, Wordperfect, Lotus, IBM, etc, que tienen los mas variados precios. Basados en el tipo de uso que se le iba a dar se escogió DaVinci Email, ya que es un producto que es ampliamente utilizado, tiene una gran integración con Novell, ha sido especialmente desarrollado para redes Novell, y es de facil uso.

Paquetes de mantenimiento de red. Estos paquetes mayormente son para control interno de la red, por ejemplo para control de impresoras, correo electrónico (para comunicación entre estaciones), para control del disco duro del 'server', etc.

Se había considerado la utilización de Windows e inclusive algunas estaciones había sido cargadas con Microsoft Windows, pero en la mayoría de los casos, pocos eran los empleados que estaban familiarizados con este tipo de paquete. Sin embargo como una siguiente fase se estaba considerando instalar este producto, ya que el largo término, existe la tendencia a desarrollar productos para este ambiente gráfico. Además en el ambiente de Windows se ha comprobado que la productividad en el uso de un computador aumenta notablemente una vez que el usuario sabe como utilizarlo.

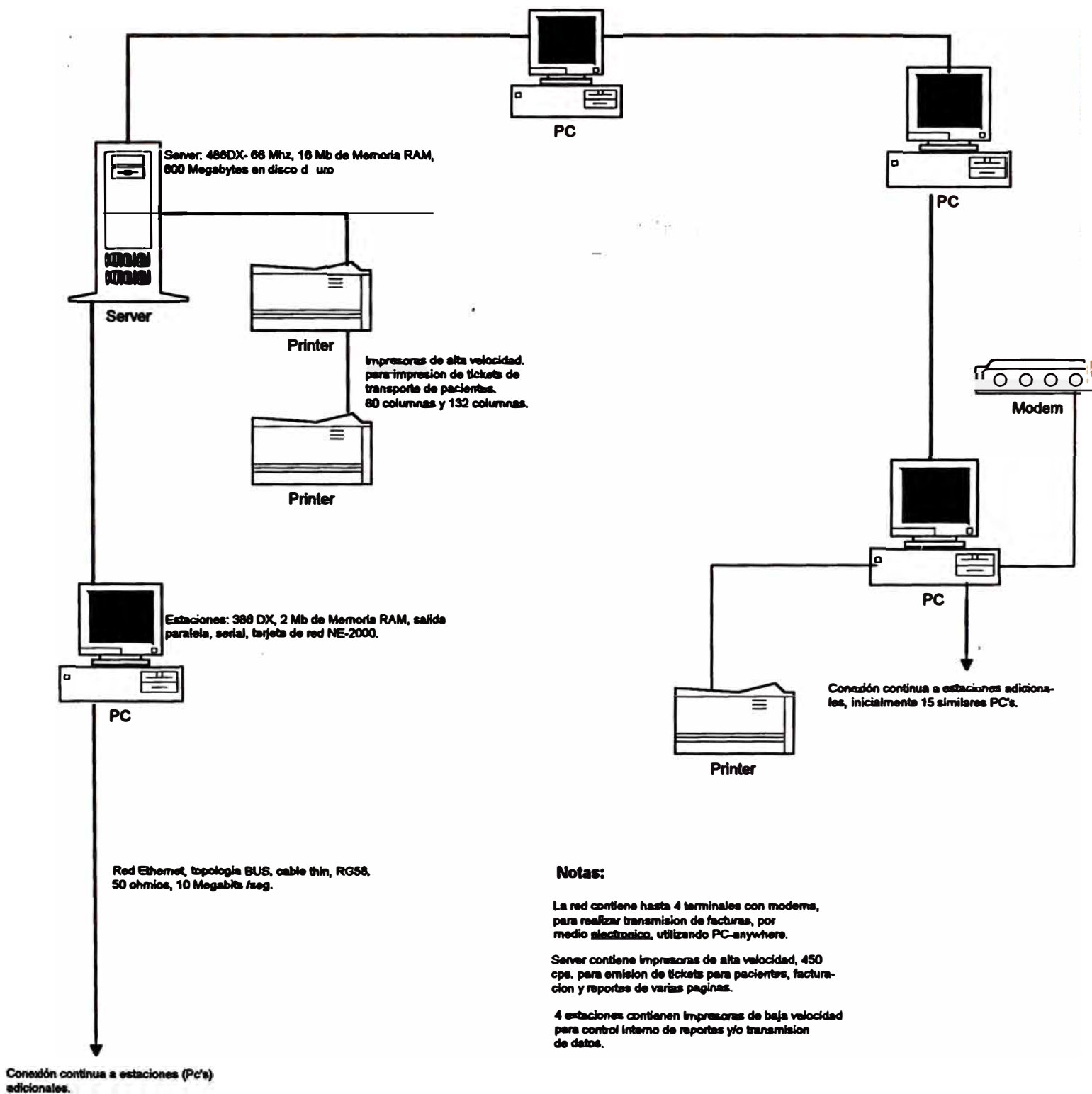


Grafico Nro.1 Arquitectura de la red

5.- Desarrollo del Software

5.1 Consideraciones Generales

En la implementación de un sistema computarizado es más importante que el hardware se adapte al software y no que el software se adapte al hardware. Es por ello que antes de comprar una pieza del sistema se haya decidido que tipo de aplicación ó aplicaciones se van a tener en la red y cuales son sus requerimientos en cuanto a hardware: si necesitan impresoras regulares ó de alta velocidad, monitores gráficos, alguna versión especial de tarjeta de red, etc.

El sistema de red se implementaba específicamente para absorber el creciente número de llamadas telefónicas que se hacía a la empresa para solicitar traslado de pacientes desde y hacia los hospitales. La aplicación a correr en la red de la empresa debería ser capaz de manejar varios terminales en línea, además poder tomar la información del paciente en ese momento (nombres, direcciones, tarjeta de asegurado, tipo de traslado, instrucciones especiales, hora de recojo, etc.).

Además esta información debe ser utilizada para elaborar las facturas y enviarlas a las diferentes hospitales y aseguradoras. Así mismo la aplicación debe poder manejar grandes volúmenes de datos ya que es necesario tener información accesible por lo menos con una antigüedad de 6 a 12 meses.

Debido a las constantes variaciones en procedimientos en las empresas de seguros y hospitales con respecto a sus asegurados y/o pacientes, el software debería tener la capacidad de poder adaptarse a estos nuevos cambios: cambios en tarifas, nuevos códigos para los asegurados, etc.. Por ejemplo un hospital tenía definidos por códigos cada uno de los servicios que prestaba a sus pacientes, con cierta regularidad estos códigos eran actualizados y/o redefinidos, y en algunos casos se refería al transporte médico, el software lógicamente debería tener la capacidad de adaptarse rápidamente a estos cambios. Igualmente debido a regulaciones gubernamentales que norman el servicio de transporte médico esta podían variar: Por ejemplo se habían regulado 3 tipos de transporte médico en ambulancia: ambulatorio, en silla de ruedas y en camilla. Cada uno de estos transportes involucra un costo y un tipo especial de ambulancia (si el paciente es ambulatorio no es necesario llevar oxígeno como cuando se transporta un paciente en camilla, por ejemplo), todos estos son requerimientos que deben ser registrados en el computador y que regularmente son revisados y cambiados y que el software debe ser capaz de tomarlos en cuenta.

5.2 Requerimientos específicos de la aplicación

Inicialmente se analizó la posibilidad de adquirir un paquete de software 'pre-hecho' pero en todos ellos se tenían que realizar cambios sustanciales para acomodarlos a la manera de trabajo que exige este tipo de servicio. La mayoría de paquetes comerciales vienen hechos para industrias que tienen mucha demanda por ejemplo comercio, servicios básicos, restaurantes, etc.

Se decidió por la elaboración de un software específicamente para 'Transportes medicos'. Se escogió Foxpro para crear la aplicación y se encargo a una empresa consultora (Technotel, Inc. de Miami, USA) el desarrollo de la aplicación. Se especificó claramente que 'Transportes medicos' debería tener el código fuente para poder realizar cambios necesarios en el software, asimismo entrenamiento de un analista-programador para el paquete a crearse. El paquete se desarrollo bajo los siguientes lineamientos

- Ingreso de la información del paciente en línea .- Datos personales, tipo de transporte requerido, clinica u hospital a ser transportado, aseguradora que solicita el servicio. Esta información básica (e información secundaria) deberá ser ingresada en el momento en que se genera el requerimiento: a travez del teléfono, via fax, requerimiento escrito. Este es el principal ingreso de información al sistema de facturación, es decir el ingreso de datos debe ser rápida, fácil de utilizar y entender por los operadores, debe ser capaz de trabajar con varios terminales a la vez.
- Acceso de la información historica con 12 meses de antigüedad .- Se debería poder acceder la información que se tenía en el computador por un año, por ejemplo facturas, pacientes transportados, servicios ofrecidos, etc. En base a esta información se iban a elaborar una serie de reportes contables (Contabilidad, estadística, análisis del servicio, transporte de ambulancias, rendimiento de los vehículos, etc.). Así como información para el gobierno, el cual exige tener registros de pacientes transportados por lo menos con 12 meses de antigüedad.
- Pre-facturación automática.- cuando una empresa aseguradora u hospital llame a 'Transportes medicos' el sistema debe ser capaz de poder estimar el costo del transporte al paciente inmediatamente (basado en distancias regulares, tipo de servicio, etc.) de esta manera la aseguradora conoce un estimado del precio a pagar antes de tomar el servicio (por supuesto el precio final no se conoce hasta que se tenga registrado la distancia exacta recorrida por la ambulancia)

- **Facturación electrónica.**- Algunas empresas aseguradoras, especialmente los hospitales del gobierno cuentan con computadores que pueden recibir las facturas directamente en sus computadores. Es decir se había pensado desarrollar la aplicación de tal modo, fuera posible enviar las facturas a los clientes via 'modem', utilizando las líneas de teléfono, los computadores se conectan entre sí y la transmisión es prácticamente instantánea. Esto representa una gran ventaja, ya que la información se envía directamente al computador del cliente. No existe el problema de impresión de las facturas, clasificación, envío al clientes, el cual a su vez debe retippear la información en sus propios computadores. El proceso de envío electrónico disminuye en por lo menos 2 semanas el proceso de pago de las facturas con los clientes.

- **Modulos comerciales regulares.** Es decir debe tener procedimientos de facturación, Cuentas por Cobrar, Cuentas por pagar. y Cuentas Corrientes. En este caso por ejemplo no es necesario un módulo de inventarios, ya que 'Transportes medicos' no maneja productos físicos. Tampoco era necesario un módulo de contabilidad, ya que esta era manejada por una empresa de contadores, sin embargo la aplicación debería tener la posibilidad de poder adaptarle modulos contables, ya que se tenía la idea de manejar internamente la contabilidad, aunque no en un futuro cercano.

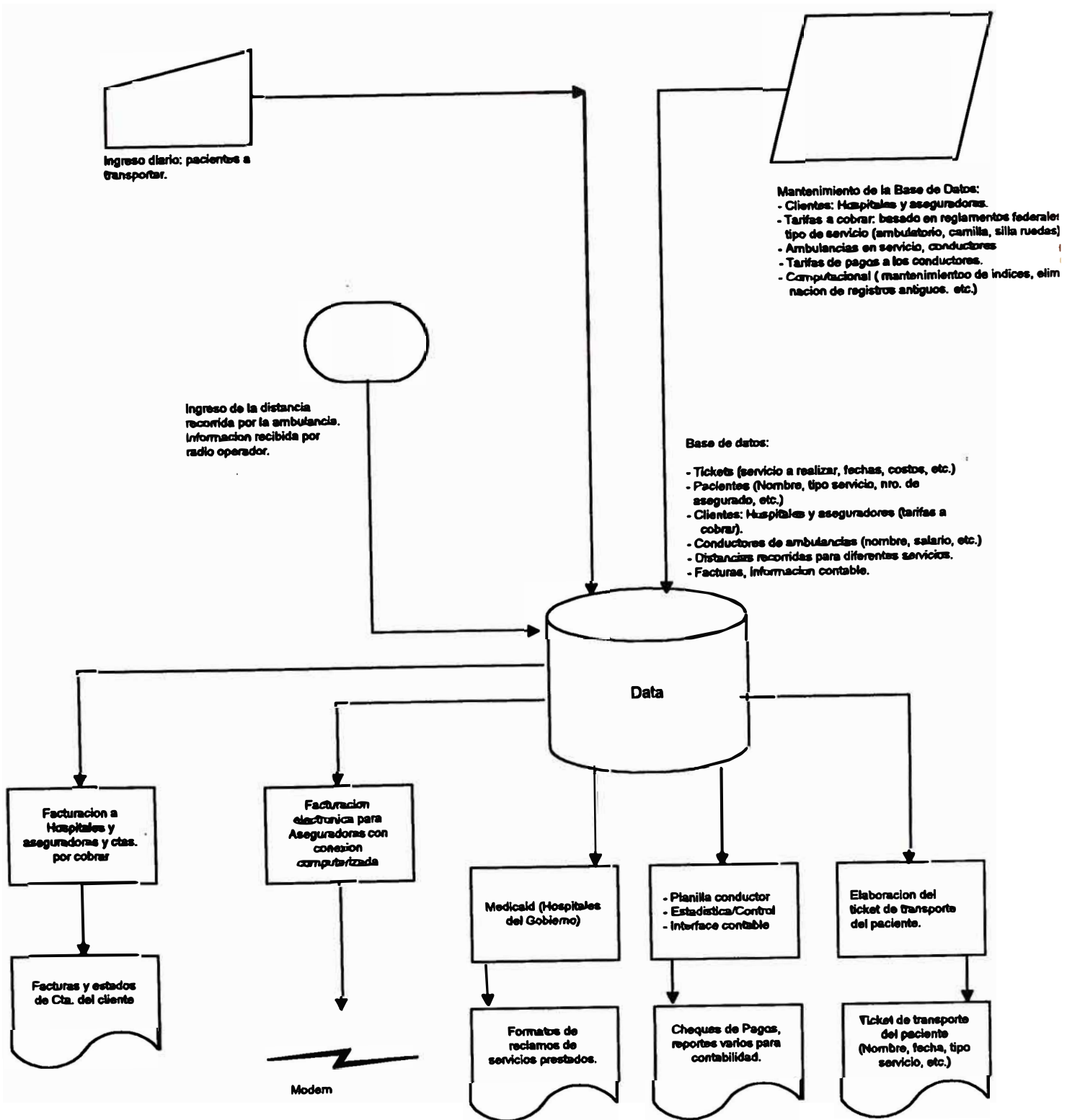
5.3 Ambiente de desarrollo - Ciclo del sistema.

El software como se dijo anteriormente fue desarrollado en Foxpro-LAN, bajo ambiente DOS, aproximadamente consta de 40000 líneas de programa, se utilizaron 250 horas-analista y 400 horas-programador para el desarrollo del software. Participaron 1 analistas y 3 programadores en el desarrollo del paquete. El paquete se desarrollo en módulos, es decir, se dividió el trabajo en partes de tal manera que cuando una parte era completada, se cargaba en la red y se podía utilizar luego de realizar la pruebas respectivas.

En el Grafico nro. 2 se muestra la arquitectura de la aplicación. La aplicación consta de varias bases de datos, todas ellas desarrolladas en FOXPRO para red. Se ha utilizado la versión 2.00 del paquete, a la fecha (Agosto '93) se esta realizando la transferencia de la aplicación a la versión 2.5.

En cuanto al ciclo de vida del sistema, se ha calculado en 6 - 7 años, es decir, el paquete debiera ser completamente cambiado ó rediseñado al término de dicho periodo; en ese período de tiempo solamente son necesarios mantenimientos regulares de la aplicación debido a cambios en la regulación del servicio de transporte medico.

Grafico Nro. 2 Arquitectura de la aplicación



5.4 Pantallas de la aplicación

En las páginas siguientes se muestran las pantallas mas importantes de la aplicación. Asi mismo se muestra un diseño en árbol de los programas mas importantes. En el anexo al final de informe se muestran los principales programas que conforman la aplicación.

En el menú principal se muestran las opciones para realizar el 'ticket trip' del paciente, es decir el documento que se emite para indicar que se debe realizar un transporte de un paciente, la cual es la opcion A. (Ver al final del informe: anexos, un ejemplo del ticket).

Del menu principal tambien se puede realizar la facturacion del transporte del paciente (la opcion B. 'Billing' del menu principal), aqui se pueden generar las facturas, clasificarlas emitirlas, modificarlas, reportes varios, etc. Tambien permite generar los archivos que se enviarian electronicamente a los clientes que pueden recibir informacion vía Modem (por telefono).

La opción C. del menu principal permite realizar un mantenimiento al archivo de clientes y sus tarifas asociadas: Crear nuevos clientes, altas, bajas, reportes, modificar su tarifas, etc.

La opción D. corresponde al modulo de cuentas por cobras, es decir, permite controlar las facturas que aun no han sido canceladas. emite los estados de cuentas para los diferentes hospitales y aseguradoras.

La opcion E. del menu principal realiza las operaciones de mantenimiento en el computador como son: chequeo de archivos, copias de seguridad, regeneracion de indices, reportes de utilizacion del disco, etc.

La opcion F corresponde a la facturacion para una cadena de hospitales llamados el CAC-Ramsay. Esta cadena de hospitales vienen a ser el cliente mas importante de la empresa y necesitan hacerle un seguimiento especial a las facturas, asi mismo permite el enlace electronico con las computadoras de ese centro hospitalario.

AMT MEDICAL TRANSP.
P.O. Box 2068
Miami, Fl 33152

- MAIN MENU
- A. Ticket trips
 - B. Billing
 - C. Customers
 - D. Receivables
 - E. Utilities
 - F. CAC-RAMSAY Claims

 - G. Exit

Use arrows to select | ← to pickup selection | ESC to cancel

To create new companies, delete, update

AMT MEDICAL TRANSP.
P.O. Box 2068
Miami, Fl 33152

- MAIN MENU
- A. Ticket trips
 - B. Billing
 - C. Customers
 - D. Receivables
 - E. Utilities
 - F. CAC-RAMSAY C1

 - G. Exit
- A. Create Tickets
 - B. Modify Tickets
 - C. Print Tickets
 - D. Delete tickets
 - E. Daily Route Sheet
 - F. Tickets report
 - G. Regular Patients file

Use arrows to select | ← to pickup selection | ESC to cancel

To Create Tickets

New Ticket Trip:

Serv. date 11/08/93

Acct. #: Bill:
 Call by: Addr:
 Phone : City: Miami St: F1 Zip:

Patient:		Caid:
Last Name:		Member #:
First Name:		Pat.Nbr.:
Pat. phone:		Auth. # :
From:		Medicare:
City: Miami	State: F1 Zip:	
To :		Time P/U Req.
City: Miami	State: F1 Zip:	0 : 0 P Am/Pm

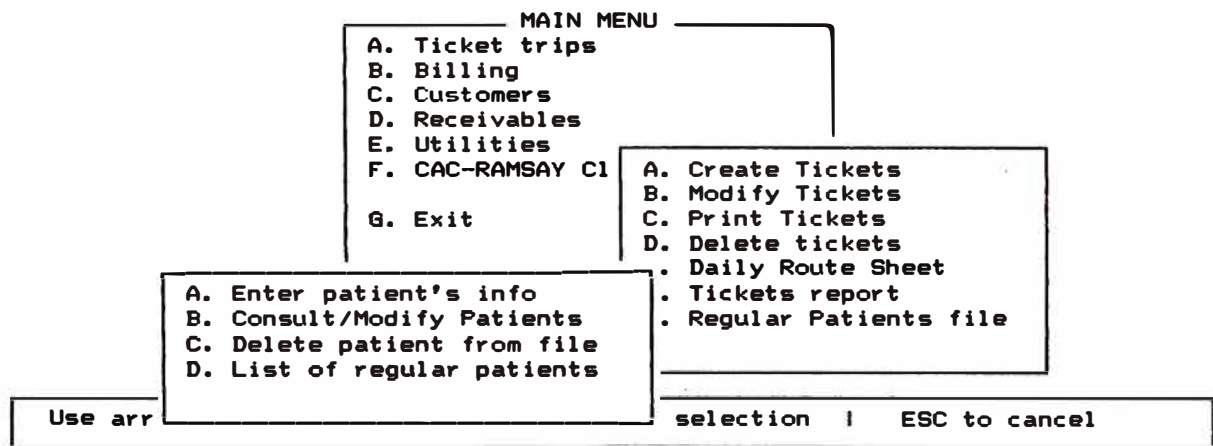
Call Taker:

1 Way/Round Trip: 1_Way Amb./1" 1Ch./Str.: Amb. Routine/AS/1: Rout.

Status:

Press F4 to get regular patients. ESC to cancel. next field

AMT MEDICAL TRANSP.
 P.O. Box 2068
 Miami, FL 33152



To Create a patient as a regular ticket

AMT MEDICAL TRANSP.
P.O. Box 2068
Miami, Fl 33152

MAIN MENU	
A. Ticket trips	
B. Billing	
C. Customers	
D. Receivables	
E. Utilities	
ims	
A. Check Invoices	
B. Print Invoices	
C. Print Medicaid	
D. Delete Invoice	
E. Summary Report	

Use arrows to select | ↵ to pickup selection | ESC to cancel

To enter mileage, additional items for invoicing

AMT MEDICAL TRANSP.

Invoice Nbr.: 10057

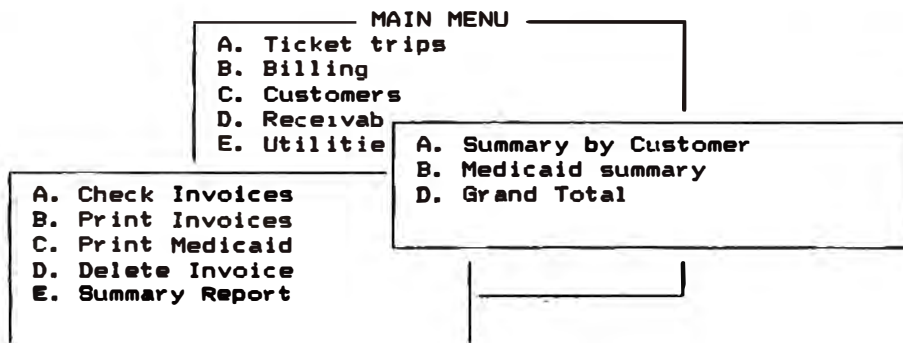
To:	Ship To:
404 - DR. SHORE - RISK MANAG	DR. SHORE - RISK MANAGMENT
Miami fl	ROAD_
Miami fl	Miami fl
Patient:	Others:
junto, lanzaon	
Fr: 783 N.W. 3 Calle To: 233 S. 45 Ave.	
Fr: 233 S. 45 Ave. To: 783 N.W. 3 Calle	

Billed: 05/15/92

Serviced: 05/04/92

Qty.	Description	Rate	Amount
2	AM001 - AMBULATORY RESPONSE CHARGE	6.00	12.00
23	AM002 - AMBULATORY RATE PER MILE	1.20	27.60
TOTAL:			39.60
ESC=Cancel PgDn=Next PgUp=Previous End=Last entered ↵ = Check			

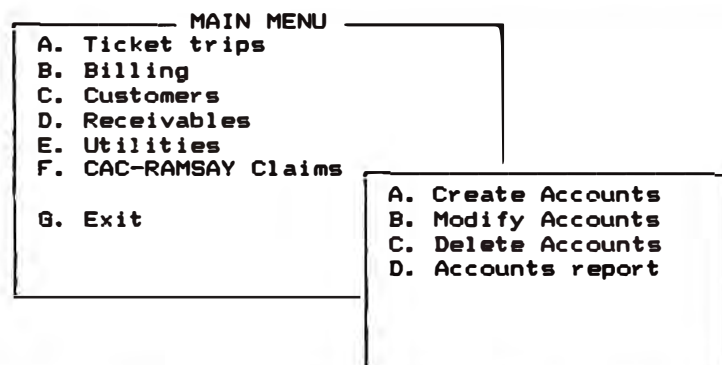
AMT MEDICAL TRANSP.
 P.O. Box 2068
 Miami, Fl 33152



Use arrows to select | ↵ to pickup selection | ESC to cancel

Report of invoices by Customer between given dates

AMT MEDICAL TRANSP.
 P.O. Box 2068
 Miami, Fl 33152



Use arrows to select | ↵ to pickup selection | ESC to cancel

To Add new Customers to the files.

AMT MEDICAL TRANSP.
 P.O. Box 2068
 Miami, Fl 33152

MAIN MENU

A. Ticket trips
 B. Billing
 C. Customers
 D. Receivables
 E. Utilities
 F. CAC-RAMSAY Claims

A. Check Payments
 B. Outstanding Invoices
 C. Summary of payments
 D. Total outstanding

Use arrows to select |] to pickup selection | ESC to cancel

To register payments received from customers

AMT MEDICAL TRANSP.
 P.O. Box 2068
 Miami, Fl 33152

Report of outstanding invoices

 You will have a list of invoices

Acct. #:
 (Type ***** if you want all of

Billed from 11/08/93 To 11/0

To Printer/Screen ? (P/S) : P

ADVANCE MEDICAL CENTER
 AMERIMED MEDICAL
 ATLAS GENERAL
 C.A.C. RAMSAY
 CAC de TEST
 CARE FLORIDA
 CHILDRENS REHAB.
 CIGNA INSURANCE
 CORAL GABLES EYE CENTER
 CORAL GABLES HOSPITAL
 CRAWFORD INS.
 DR. CHARLES BYRD
 DR. KERN BRAD

Us

Arrows to move, or first letter.] to select

AMT MEDICAL TRANSP.
P.O. Box 2068
Miami, Fl 33152

MAIN MENU	
A. Ticket trips	
B. Billing	
C. Customers	
D. Receivables	
E. Utilities	
F. CAC-RAMSAY Claims	
G. Exit	A. Prepare Billing claims B. Create Billing file C. Reports

Use arrows to select | ↵ to pickup selection | ESC to cancel

To select claims by Invoices or by range-date

AMT MEDICAL TRANSP.
P.O. Box 2068
Miami, Fl 33152

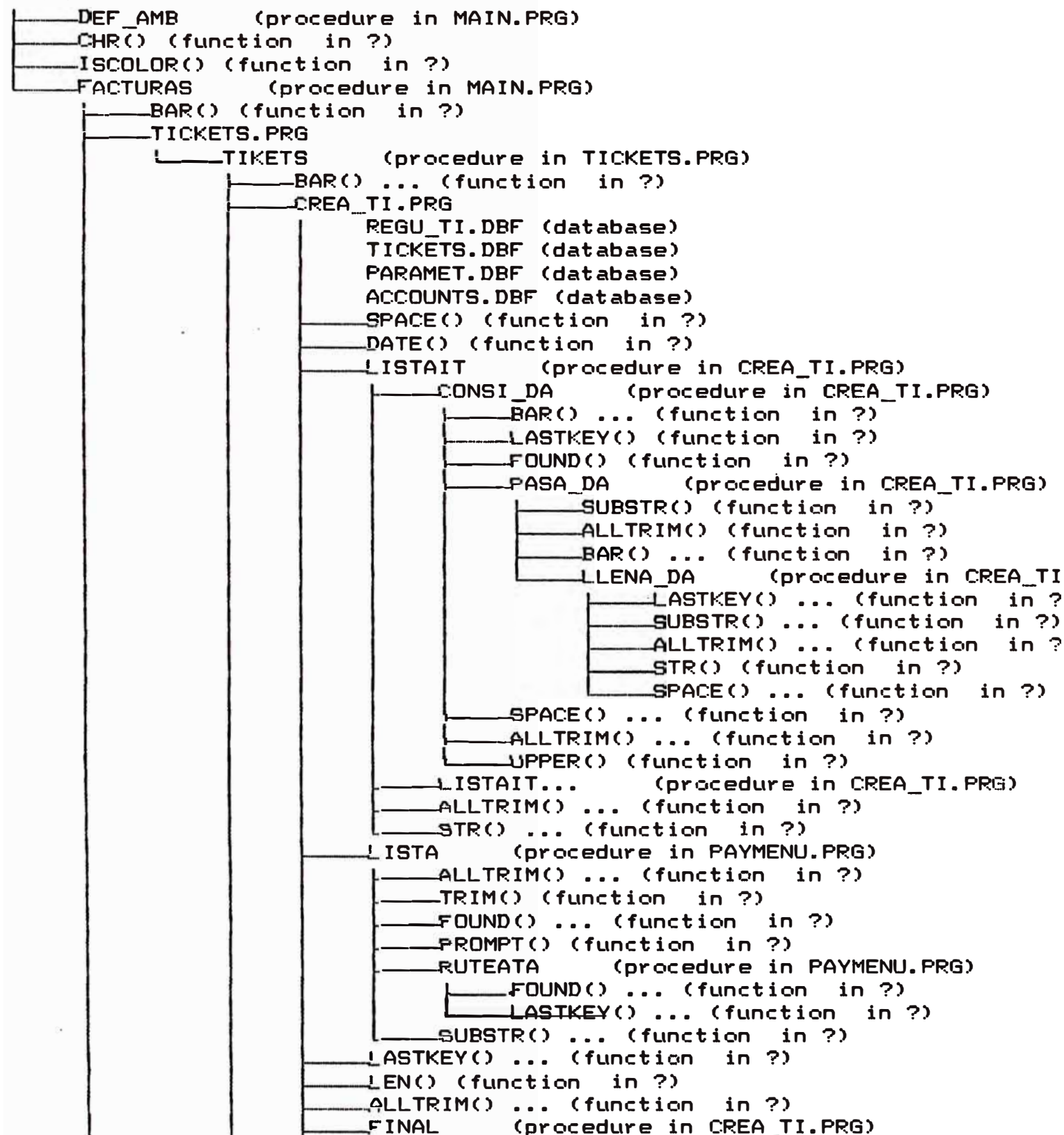
MAIN MENU	
A. Ticket trips	
B. Billing	
C. Customers	
D. Receivables	
E. Utilities	
F. CAC-RAMSAY Claims	
G. Exit	A. Prepare Billing claims B. Create Billing file C. Reports
A. Claims billed to CAC-RAMSAY B. Claims ready to send C. Uncomplete claims	

Use a | p selection | ESC to cancel

Invoices Billed to CAC-RAMSAY

System: TRANSPORTES MEDICOS
 Author: TECHNOTEL, INC.
 10/21/93 15:04:18
 Tree Diagram

MAIN.PRG



```

    ___PAD() (function in ?)
    GRABA_TI (procedure in CREA_TI.PRG)
    ___FLOCK() (function in ?)
    ___SUBSTR() ... (function in ?)
    ___DATE() ... (function in ?)
    ___TIME() (function in ?)
    ___STR() ... (function in ?)
    TICK_PR.PRG
    K:\MEDICAL\TICKET.FRX (report form)
    K:\MEDICAL\TICKET.FRX (report form)
    ___SYS() (function in ?)
    GENE_ACT (procedure in CREA_TI.PRG)
    ___LEN() ... (function in ?)
    ___ALLTRIM() ... (function in ?)
    ___TYPE() (function in ?)
    ___SUBSTR() ... (function in ?)
    ___VAL() (function in ?)
    ___FLOCK() ... (function in ?)
    ___STR() ... (function in ?)
    MODITI.PRG
    TICKETS.DBF (database)
    PARAMET.DBF (database)
    ACCOUNTS.DBF (database)
    ___DATE() ... (function in ?)
    ___SPACE() ... (function in ?)
    TITULOS (procedure in MODITI.PRG)
    ___ISCOLOR() ... (function in ?)
    ___LASTKEY() ... (function in ?)
    ___BOF() (function in ?)
    SHOW_TICKET (procedure in MODITI.PRG)
    ___ISCOLOR() ... (function in ?)
    ___ALLTRIM() ... (function in ?)
    ___FOUND() ... (function in ?)
    ___STR() ... (function in ?)
    ___SUBSTR() ... (function in ?)
    ___EOF() (function in ?)
    ___FOUND() ... (function in ?)
    TOMA_VAL (procedure in MODITI.PRG)
    ___SPACE() ... (function in ?)
    SHOW_TICKET... (procedure in MODITI.PRG)
    ___ALLTRIM() ... (function in ?)
    ___LASTKEY() ... (function in ?)
    ___RECCOUNT() (function in ?)
    ___DATE() ... (function in ?)
    ___STR() ... (function in ?)
    LISTA... (procedure in PAYMENU.PRG)
    ___LEN() ... (function in ?)
    ___ALLTRIM() ... (function in ?)
    FINAL... (procedure in CREA_TI.PRG)
    MODI_TI (procedure in MODITI.PRG)
    ___FLOCK() ... (function in ?)
    ___SUBSTR() ... (function in ?)
    TICK_PR.PRG...
    PRITICK.PRG
    TICKETS.DBF (database)
    PARAMET.DBF (database)
    ACCOUNTS.DBF (database)

```


- DATE() ... (function in ?)
- SPACE() ... (function in ?)
- TITULOS... (procedure in MODITI.PRG)
- LASTKEY() ... (function in ?)
- BOF() ... (function in ?)
- SHOW_TICKET... (procedure in MODITI.PRG)
- EOF() ... (function in ?)
- FOUND() ... (function in ?)
- TOMA_VAL... (procedure in MODITI.PRG)
- STR() ... (function in ?)
- FINAL... (procedure in CREA_TI.PRG)
- TICK_PR.PRG...
- TICK_PR2.PRG
 - PARAMET.DBF (database)
 - K:\MEDICAL\TICKET2.FRX (report form)
 - K:\MEDICAL\TICKET2A.FRX (report form)
 - SYS() ... (function in ?)

DELETI.PRG

- TICKETS.DBF (database)
- PARAMET.DBF (database)
- ACCOUNTS.DBF (database)
- DATE() ... (function in ?)
- SPACE() ... (function in ?)
- TITULOS... (procedure in MODITI.PRG)
- LASTKEY() ... (function in ?)
- BOF() ... (function in ?)
- SHOW_TICKET... (procedure in MODITI.PRG)
- EOF() ... (function in ?)
- FOUND() ... (function in ?)
- TOMA_VAL... (procedure in MODITI.PRG)
- STR() ... (function in ?)
- FINAL... (procedure in CREA_TI.PRG)

DAY_ROUT.PRG

- OPTI (procedure in DAY_ROUT.PRG)
 - BAR() ... (function in ?)
 - ROUTES.PRG
 - TICKETS.DBF (database)
 - PARAMET.DBF (database)
 - ROUTES.DBF (database)
 - &WFILE (database)
 - ORDEN (index file)
 - &WINDI (index file)
 - SPACE() ... (function in ?)
 - LASTKEY() ... (function in ?)
 - STR() ... (function in ?)
 - FILE() (function in ?)
 - BOF() ... (function in ?)
 - SHOW_TICKET... (procedure in MODITI.PRG)
 - EOF() ... (function in ?)
 - RECCOUNT() ... (function in ?)
 - DELETED() (function in ?)
 - FOUND() ... (function in ?)
 - TOMA_VAL... (procedure in MODITI.PRG)
 - REP_ROUT.PRG
 - &WFILE (database)
 - TICKETS.DBF (database)
 - &WINDI (index file)

```

LASTKEY() ... (function in ?)
STR() ... (function in ?)
FILE() ... (function in ?)
TITULO      (procedure in REP_ACCT.PR
    DATE() ... (function in ?)
    STR() ... (function in ?)
    REPLICATE() (function in ?)
EOF() ... (function in ?)
FOUND() ... (function in ?)
SUBSTR() ... (function in ?)
ALLTRIM() ... (function in ?)
LASTKEY() ... (function in ?)
STR() ... (function in ?)
FILE() ... (function in ?)
REP_ROUT.PRG...
TI_REPO.PRG
    TICKETS.DBF (database)
    K:\MEDICAL\TIEMIT.FRX (report form)
    K:\MEDICAL\TISERV.FRX (report form)
    DATE() ... (function in ?)
    LASTKEY() ... (function in ?)
REGUPA.PRG
    REGPATS      (procedure in REGUPA.PRG)
    BAR() ... (function in ?)
    CREA_PA.PRG
        REGU_TI.DBF (database)
        PARAMET.DBF (database)
        ACCOUNTS.DBF (database)
        SPACE() ... (function in ?)
        DATE() ... (function in ?)
        LISTA...    (procedure in PAYMENU.PRG)
        LASTKEY() ... (function in ?)
        LEN() ... (function in ?)
        ALLTRIM() ... (function in ?)
        FINAL...    (procedure in CREA_TI.PRG)
        GRABA_TI... (procedure in CREA_TI.PRG)
    MODI_PA.PRG
        REGU_TI.DBF (database)
        PARAMET.DBF (database)
        ACCOUNTS.DBF (database)
        DATE() ... (function in ?)
        SPACE() ... (function in ?)
        TITULOS...  (procedure in MODITI.PRG)
        LASTKEY() ... (function in ?)
        BOF() ... (function in ?)
        SHOW_TICKET... (procedure in MODITI.P
        EOF() ... (function in ?)
        FOUND() ... (function in ?)
        TOMA_VAL... (procedure in MODITI.PRG)
        STR() ... (function in ?)
        LISTA...    (procedure in PAYMENU.PRG)
        LEN() ... (function in ?)
        ALLTRIM() ... (function in ?)
        FINAL...    (procedure in CREA_TI.PRG)
        MODI_TI...  (procedure in MODITI.PRG)
        TICK_PR.PRG...

```

```

DELE_PA.PRG
  REGU_TI.DBF (database)
  PARAMET.DBF (database)
  ACCOUNTS.DBF (database)
  DATE() ... (function in ?)
  SPACE() ... (function in ?)
  TITULOS... (procedure in MODITI.PRG)
  LASTKEY() ... (function in ?)
  BOF() ... (function in ?)
  SHOW_TICKET... (procedure in MODITI.P)
  EOF() ... (function in ?)
  FOUND() ... (function in ?)
  STR() ... (function in ?)
  FINAL... (procedure in CREA_TI.PRG)
LIST_RE.PRG
  TICKETS.DBF (database)
  REGU_TI.DBF (database)
  K:\MEDICAL\PR_REGU.FRX (report form)
  DATE() ... (function in ?)
  LASTKEY() ... (function in ?)
MENUBILL.PRG
  BILLS (procedure in MENUBILL.PRG)
  BAR() ... (function in ?)
  CHK_INVO.PRG
    REGU_TI.DBF (database)
    TICKETS.DBF (database)
    PARAMET.DBF (database)
    ACCOUNTS.DBF (database)
    RATES.DBF (database)
    SERVICES.DBF (database)
    INVOICES.DBF (database)
    FINAL... (procedure in CREA_TI.PRG)
    TITULOS... (procedure in MODITI.PRG)
    LASTKEY() ... (function in ?)
    BOF() ... (function in ?)
    SHOW_INVO (procedure in CHK_INVO.PRG)
      ISCOLOR() ... (function in ?)
      CREA_INVO (procedure in CHK_INVO.PRG)
        FOUND() ... (function in ?)
        ALLTRIM() ... (function in ?)
        STR() ... (function in ?)
      ALLTRIM() ... (function in ?)
      FOUND() ... (function in ?)
      LEN() ... (function in ?)
      EOF() ... (function in ?)
      ROUND() (function in ?)
      TITULOS... (procedure in MODITI.PRG)
    EOF() ... (function in ?)
    FOUND() ... (function in ?)
    MILES (procedure in CHK_INVO.PRG)
      &WFILEDBF (database)
      &WARCHI (database)
      WARCHI.DBF (database)
      &WARCHI (index file)
      DATE() ... (function in ?)
      LASTKEY() ... (function in ?)
      STR() ... (function in ?)

```

- LEN() ... (function in ?)
- ALLTRIM() ... (function in ?)
- FOUND() ... (function in ?)
- RAND() (function in ?)
- EOF() ... (function in ?)
- LISTA... (procedure in PAYMENU.PRG)
- SHOW_INVO... (procedure in CHK_INVO.PRG)
- OTRO_ITEM (procedure in CHK_INVO.PRG)

PR_INVO.PRG

- TICKETS.DBF (database)
- SERVICES.DBF (database)
- INVOICES.DBF (database)
- DATE() ... (function in ?)
- LASTKEY() ... (function in ?)
- SETEA (procedure in PR_INVO.PRG)
- TEST_INVO (procedure in PR_INVO.PRG)
- EOF() ... (function in ?)
- LEN() ... (function in ?)
- ALLTRIM() ... (function in ?)
- PR_FORMA (procedure in PR_INVO.PRG)
 - STR() ... (function in ?)
 - ALLTRIM() ... (function in ?)
 - LEN() ... (function in ?)
 - FOUND() ... (function in ?)
 - ROUND() ... (function in ?)

PR_CAID.PRG

- TICKETS.DBF (database)
- SERVICES.DBF (database)
- INVOICES.DBF (database)
- DATE() ... (function in ?)
- LASTKEY() ... (function in ?)
- TEST_FORM (procedure in PR_CAID.PRG)
 - DATE() ... (function in ?)
- EOF() ... (function in ?)
- LEN() ... (function in ?)
- ALLTRIM() ... (function in ?)
- PR_HRS (procedure in PR_CAID.PRG)
 - ALLTRIM() ... (function in ?)
 - FOUND() ... (function in ?)
 - ROUND() ... (function in ?)
 - DATE() ... (function in ?)

DELEIN.PRG

- TICKETS.DBF (database)
- PARAMET.DBF (database)
- ACCOUNTS.DBF (database)
- RATES.DBF (database)
- SERVICES.DBF (database)
- INVOICES.DBF (database)
- FINAL... (procedure in CREA_TI.PRG)
- TITULOS... (procedure in MODITI.PRG)
- LASTKEY() ... (function in ?)
- BOF() ... (function in ?)
- SHOW_INVO... (procedure in CHK_INVO.PRG)
- EOF() ... (function in ?)
- FOUND() ... (function in ?)
- ELIMINA (procedure in DELEIN.PRG)
 - FOUND() ... (function in ?)
 - LASTKEY() ... (function in ?)

```

└── EOF() ... (function in ?)
└── PRSALES.PRG
    └── REPORTES (procedure in PRSALES.PRG)
        TICKETS.DBF (database)
        PARAMET.DBF (database)
        ACCOUNTS.DBF (database)
        BAR() ... (function in ?)
        BY_CUST (procedure in PRSALES.PRG)
            K:\MEDICAL\BY_ACCT.FRX (report form)
            SPACE() ... (function in ?)
            DATE() ... (function in ?)
            LISTA... (procedure in PAYMENU.PRG)
            LASTKEY() ... (function in ?)
            ALLTRIM() ... (function in ?)
        PRCAID (procedure in PRSALES.PRG)
            K:\MEDICAL\BY_CAID.FRX (report form)
            SPACE() ... (function in ?)
            DATE() ... (function in ?)
            LASTKEY() ... (function in ?)
            LEN() ... (function in ?)
            ALLTRIM() ... (function in ?)
        GRAND_TO (procedure in PRSALES.PRG)
            K:\MEDICAL\TOT_ACCT.FRX (report form)
            SPACE() ... (function in ?)
            DATE() ... (function in ?)
            LASTKEY() ... (function in ?)
└── ACCOUNTS.PRG
    └── OPTIONES (procedure in ACCOUNTS.PRG)
        BAR() ... (function in ?)
        ADDACCT.PRG
            RATES.DBF (database)
            ACCOUNTS.DBF (database)
            &FICHERO (database)
            SERVICES.DBF (database)
            &WNOBMBRE (database)
            STR() ... (function in ?)
            RAND() ... (function in ?)
            FILE() ... (function in ?)
            LASTKEY() ... (function in ?)
            LEN() ... (function in ?)
            ALLTRIM() ... (function in ?)
            FOUND() ... (function in ?)
            FINAL... (procedure in CREA_TI.PRG)
        MOD_ACCT.PRG
            ACCOUNTS.DBF (database)
            RATES.DBF (database)
            SERVICES.DBF (database)
            &WFILE2 (database)
            SPACE() ... (function in ?)
            TITULOS... (procedure in MODITI.PRG)
            LASTKEY() ... (function in ?)
            EOF() ... (function in ?)
            SHOW_ACCT (procedure in MOD_ACCT.PRG)
                └── ISCOLOR() ... (function in ?)
            EOF() ... (function in ?)
            FOUND() ... (function in ?)
            TOMA_VALA (procedure in MOD_ACCT.PRG)

```

```

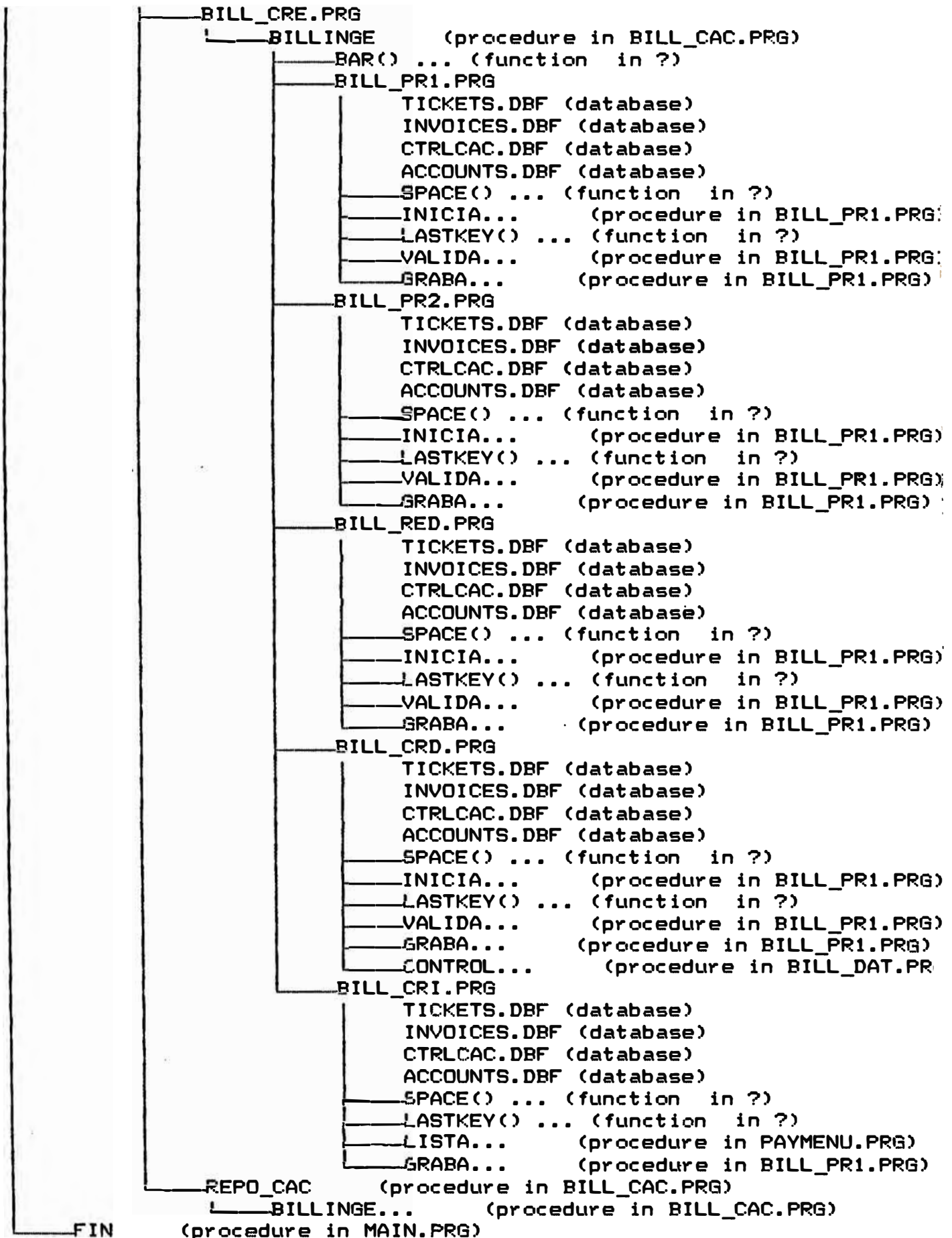
      |
      |---LEN() ... (function in ?)
      |---ALLTRIM() ... (function in ?)
      |---STR() ... (function in ?)
      |---RAND() ... (function in ?)
      |---FILE() ... (function in ?)
      |---FINAL... (procedure in CREA_TI.PRG)
      |---REP_ACCT.PRG
      |   |---RATES.DBF (database)
      |   |---ACCOUNTS.DBF (database)
      |   |---SERVICES.DBF (database)
      |   |---LASTKEY() ... (function in ?)
      |   |---TITULO... (procedure in REP_ACCT.PRG)
      |   |---EOF() ... (function in ?)
      |   |---ALLTRIM() ... (function in ?)
      |   |---FOUND() ... (function in ?)
      |---PAYMENU.PRG
      |   |---PAYABS (procedure in PAYMENU.PRG)
      |   |---TICKETS.DBF (database)
      |   |---PARAMET.DBF (database)
      |   |---ACCOUNTS.DBF (database)
      |   |---BAR() ... (function in ?)
      |   |---CHK_PAY.PRG
      |   |   |---TICKETS.DBF (database)
      |   |   |---PARAMET.DBF (database)
      |   |   |---ACCOUNTS.DBF (database)
      |   |   |---RATES.DBF (database)
      |   |   |---SERVICES.DBF (database)
      |   |   |---INVOICES.DBF (database)
      |   |   |---TITULOS... (procedure in MODITI.PRG)
      |   |   |---LASTKEY() ... (function in ?)
      |   |   |---BOF() ... (function in ?)
      |   |   |---SHOW_INVO... (procedure in CHK_INVO.PRG)
      |   |   |---EOF() ... (function in ?)
      |   |   |---FOUND() ... (function in ?)
      |   |   |---PAGO (procedure in CHK_PAY.PRG)
      |   |   |   |---STR() ... (function in ?)
      |   |   |   |---LASTKEY() ... (function in ?)
      |   |---PR_PAY (procedure in PAYMENU.PRG)
      |   |   |---K:\MEDICAL\PY_ACCT.FRX (report form)
      |   |   |---SPACE() ... (function in ?)
      |   |   |---DATE() ... (function in ?)
      |   |   |---LISTA... (procedure in PAYMENU.PRG)
      |   |   |---LASTKEY() ... (function in ?)
      |   |   |---ALLTRIM() ... (function in ?)
      |   |---PR_CANCEL (procedure in PAYMENU.PRG)
      |   |   |---K:\MEDICAL\PY_ACCT.FRX (report form)
      |   |   |---SPACE() ... (function in ?)
      |   |   |---DATE() ... (function in ?)
      |   |   |---LISTA... (procedure in PAYMENU.PRG)
      |   |   |---LASTKEY() ... (function in ?)
      |   |   |---ALLTRIM() ... (function in ?)
      |   |---PR_DUES (procedure in PAYMENU.PRG)
      |   |   |---K:\MEDICAL\PY_ACCT1.FRX (report form)
      |   |   |---SPACE() ... (function in ?)
      |   |   |---DATE() ... (function in ?)
      |   |   |---LASTKEY() ... (function in ?)
      |---PARAMETS.PRG
      |   |---PARAMT (procedure in PARAMETS.PRG)
  
```

```

BAR() ... (function in ?)
UTILIT.PRG
  TICKETS.DBF (database)
  &WFILE (database)
  INVOICES.DBF (database)
  LASTKEY() ... (function in ?)
  DATE() ... (function in ?)
  STR() ... (function in ?)
  RAND() ... (function in ?)
BILL_CAC.PRG
  BILLINGC (procedure in BILL_CAC.PRG)
  BAR() ... (function in ?)
  BILL_SEL.PRG
    BILLINGS (procedure in BILL_SEL.PRG)
      BAR() ... (function in ?)
      BILL_DAT.PRG
        TICKETS.DBF (database)
        INVOICES.DBF (database)
        CTRLCAC.DBF (database)
        ACCOUNTS.DBF (database)
        SPACE() ... (function in ?)
        INICIA (procedure in BILL_PR1.PRG)
          CF916486.DBF (database)
          LISTA... (procedure in PAYMENU.P
          LASTKEY() ... (function in ?)
          SPACE() ... (function in ?)
          RECCOUNT() ... (function in ?)
          DATE() ... (function in ?)
          SUBSTR() ... (function in ?)
          STR() ... (function in ?)
          ALLTRIM() ... (function in ?)
          LEN() ... (function in ?)
          REPLICATE() ... (function in ?)
          CTOD() (function in ?)
          DTOC() (function in ?)
          VAL() ... (function in ?)
          FILE() ... (function in ?)
          FOUND() ... (function in ?)
          LASTKEY() ... (function in ?)
          VALIDA (procedure in BILL_PR1.PRG)
            FOUND() ... (function in ?)
            EOF() ... (function in ?)
            CARGA (procedure in BILL_PR1.PRG)
            LASTKEY() ... (function in ?)
          GRABA (procedure in BILL_PR1.PRG)
            TICKTEMR.DBF (database)
            C0916486.DBF (database)
            CF916486.DBF (database)
            TICKTEMR.IDX (index file)
            C0916486.IDX (index file)
            SPACE() ... (function in ?)
            ALLTRIM() ... (function in ?)
            FILE() ... (function in ?)
            EOF() ... (function in ?)
            FOUND() ... (function in ?)
            RECCOUNT() ... (function in ?)
            PRTCAC (procedure in BILL_PR1.PRG)
            RECNO() (function in ?)

```

1. PRG)						— PAGEBREAK (procedure in BIL
L_PR1. PRG)						— EOF() ... (function in ?)
in BILL_PR1. PRG)						— PAGEHEAD (procedure in
in ?)						— PAGETITLE (proced
in ?)						— SYS() ... (funct
dure in BILL_PR1. PRG)						— DATE() ... (func
tion in ?)						— PRTCENTER (p
n in ?)						— LEN() ... (
on in ?)						— LEFT() (fun
PR1. PRG)						— EMPTY() (fu
RG)						— PAGEHEAD... (procedure in B
?)						— DETAIL (procedure in BILL_P
						— ALLTRIM() ... (function i
						— LEFT() ... (function in ?
						— LEN() ... (function in ?)
						— SPACE() ... (function in
						— ROUND() ... (function in
						— REPLICATE() ... (function
						— SUBSTR() ... (function in
						— LEN() ... (function in ?)
						— TYPE() ... (function in ?)
						— SUBSTR() ... (function in ?)
						— REPLICATE() ... (function in ?)
						— LEFT() ... (function in ?)
						— STR() ... (function in ?)
						— AT() (function in ?)
						— RECNO() ... (function in ?)
						— DTOC() ... (function in ?)
						— ROUND() ... (function in ?)
						CONTROL (procedure in BILL_DAT. PRG)
						C0916486. DBF (database)
						— FOUND() ... (function in ?)
						— EOF() ... (function in ?)
						— CARGA... (procedure in BILL_PR1. PRG)
						— VAL() ... (function in ?)
						BILL_INV. PRG
						TICKETS. DBF (database)
						INVOICES. DBF (database)
						CTRLCAC. DBF (database)
						ACCOUNTS. DBF (database)
						— SPACE() ... (function in ?)
						— LISTA... (procedure in PAYMENU. PRG)
						— LASTKEY() ... (function in ?)
						— INICIA... (procedure in BILL_PR1. PRG)
						— GRABA... (procedure in BILL_PR1. PRG)



6. Recursos Humanos

6.1 Para la instalación de la red

Se utilizaron 3 tecnicos que realizaron el cableado, instalación del sistema operativo Novell en el server, configurar y conectar las estaciones, impresoras, modems. Este trabajo tomó alrededor de 3 semanas. Probablemente la parte mas dificil es configurar el 'server' con el sistema operativo Novell, ya que exige un buen conocimiento del hardware en el computador (tipo de tarjeta a utilizar, si existen discos duros externos, configuracion de impresoras, conexiones a otros 'servers', etc.).

Para el software, como se dijo anteriormente se utilizó un equipo de 4 personas para el desarrollo del software, esta aplicación es la columna principal en la cual se iba a desarrollar la actividad de 'Transportes medicos': proveer servicio de transporte a los pacientes; sin embargo se habian planeado instalar versiones de red de paquetes comerciales de uso conocido como son:

Lotus 123 .- para el desarrollo de cuadros estadísticos

Wordperfect .- Para el procesamiento de textos: cartas, estimados, etc.

PCanywhere.- para la transmisión de información vía 'modem' a los diferentes hospitales y aseguradoras sobre el transporte de sus pacientes.

6.2 Personal Operativo

Una vez instalado los diferentes paquetes en la red, solamente se hizo un reentrenamiento en el paquete especialmente desarrollado para la empresa, ya que los otros paquetes como Lotus y Wordperfect habian sido utilizados anteriormente por los empleados de la empresa.

Con el nuevo paquete, ahora solamente se necesitan cuatro personas para ingresar los datos, se hace mas rapida y eficientemente que antes. Solamente toma alrededor de 2 minutos para ingresar la informacion completa del paciente y poder dar un estimado del costo de transportar un paciente.

6.3 Mantenimiento de la aplicación

Así mismo 'Transportes medicos' había contratado a la misma empresa consultora para realizar los cambios que sean necesarios en el paquete para que se adapte a los probables cambios que se pudieran presentar. En un principio la empresa penso en contratar un analista-programador para realizar el mantenimiento de la aplicacion, pero resultaba mas economico contratar la empresa consultora para realizar los posibles cambios, que en promedio se podría presentar una vez cada 2 a 3 meses y no involucraba mas de 1 semana de trabajo.

Sin embargo se asigno un 'administrador de red', ya que el tamaño de la red justificaba que existiera una persona con la suficiente conocimiento en computadoras para realizar trabajos menores en el mantenimiento de la red. La empresa consultora reentreno al asistente de la Gerencia general en la administracion básica de la red (por ejemplo agregar un nuevo usuario a la red, redireccionar impresoras, conectar y desconectar la red, realizar copias diarios de seguridad, etc.), el mantenimiento mas detallado era realizado por la empresa consultora cuando era necesario (cambio de equipos dañados, configuracion de nuevo software, etc.)

7.- Puesta en marcha de la aplicación.

7.1 Trabajo en paralelo

En la implantación de un nuevo sistema computarizado el cambio no se puede dar súbitamente. Se realizó un 'paralelo', es decir gradualmente se fueron incorporando las diferentes funciones computarizadas y cuando estuvieran bien entendidas se abandonaba el sistema manual.

Con esta idea en mente una vez instalado el hardware y probado los diferentes módulos, se inicio un 'paralelo' con la toma de ordenes de los pacientes, es decir antes esta operacion se realizaba en forma manual, 2 empleados que realizaban esta labor fueron reentrenados para realizar la misma funcion via computador, el resto de empleados continuaba tomando las ordenes manualmente. Cuando se verificaba que la toma de ordenes era entendido por los demas empleados y se verificaba la consistencia de datos se automatizaba totalmente esa labor y los empleados eran reentrenados, en este caso ya no eran necesarios tantos empleados, se redujeron a cuatro personas la labor de toma de ordenes con el computador.

La automatización de los procesos seguía el flujo de control de los documentos, es decir. en donde se creaba el documento se iniciaba la automatización (en este caso toma de órdenes). La toma de una orden, el cual consistia de la informacion del paciente, tipo de transportes, numero de seguro, hora de recogo, etc. generaba lo que se llama un 'ticket'. Luego el ticket era programado para ser asignado a una ambulancia de acuerdo a la hora y día de recojo.

Cuando el paciente era recogido por la ambulancia y transportado al lugar asignado el chofer de la ambulancia se comunicaba por radio con 'Transportes Medicos' para indicar la hora y distancia recorrida, la cual era anotado por el operador de radio. Este fue el siguiente paso que se automatizó: el ingreso de la distancia recorrida ahora era ingresada en el computador.

El siguiente paso fue automatizar la generación de las facturas para las diferentes hospitales y empresas de seguros. Cuando se realizaba manualmente, la facturas eran generadas en un computador personal (no en red) con un paquete comercial con limitadas funciones de facturacion (se tenia que digitar varias veces un mismo datos, la lista de pacientes que podia aceptar era limitada, el paquete no habia sido diseñado para este tipo de servicio, etc.).

Ahora la facturación en la red era generada automáticamente ya que se tenían los pacientes en el computador, y la distancia recorrida, basada en las tarifas de cobro a los hospitales y aseguradoras se obtenía la factura.

7.2 Orden de automatización

El proceso de realizar el cambio de la forma manual al procedimiento computacional tomó aproximadamente 45 días, cuando se corría en paralelo alguna función y se encontraba algún problema, entonces no se continuaba hasta resolver completamente el problema. Los problemas que se encontraron fueron mayormente operativos, no computacionales. El orden que se tuvo en la automatización fue:

- Entrada de datos del paciente.- Información del paciente a transportar : direcciones, empresa de seguros u hospital que solicita el servicio, tipo de transporte (Camilla, silla de ruedas, ambulatorio), hora y fecha de recojo, etc.
- Módulo para operadores de radio.- Una vez transportado el paciente, los operadores de radio digitan la distancia recorrida por la ambulancia, el computador almacena esta información para elaborar las facturas.
- Facturación.- Generación de facturas a los diferentes empresas o aseguradoras.
- Cuentas por Cobrar.- Registro de los pagos efectuados por los hospitales a 'Transportes médicos'
- Módulos y reportes contables.- generación de reportes para la empresa que manejaba la contabilidad de 'Transportes médicos' : Libro diario, Libro mayor, reportes de las diferentes cuentas.
- Módulo para control de Ambulancias.- Controla y registra las diferentes ambulancias y personal que manejan las ambulancias, esta información es utilizada para elaborar planilla de pagos.
- Transferencia electrónica .- permite enviar las facturas directamente a los computadores de los hospitales o empresas de seguros para obtener más rápidamente el pago de las facturas generadas.

8. Conclusiones

8.1 Conclusiones Generales

La implantación de un sistema de red era un paso que tenía que darse, a fin de absorber la demanda en la toma de órdenes para los pacientes, el no haberlo hecho hubiera costado a la empresa el tener que triplicar el personal empleado. El sistema computarizado si bien es cierto demanda una inicial inversión sustancial, es inmediatamente recuperada con el ahorro que se obtiene debido a productividad que ofrece.

El costo en general de transportar un paciente se había reducido ahora en un 25% debido a ahorro de recursos humanos: con el mismo personal y solamente aumento de ambulancias era posible transportar mayor cantidad de pacientes a un menor costo.

El ciclo de vida de la aplicación era de 7 años, es decir al cabo de ese tiempo se tiene que revisar completamente la aplicación y reemplazarla por otra, asumiendo un crecimiento en la empresa de 20 % en incremento de transporte de pacientes por año.

Reentrenamiento de personal. Los empleados de la empresa se encontraban ahora con una mejor preparación en computadores, esto se demostraba en el aumento de productividad, 'Transportes medicos' prácticamente no tuvo que contratar nuevos empleados para absorber la nueva demanda, solamente se reasignaron posiciones.

8.2 Respecto a la aplicación

Luego de instalada la aplicación, la empresa 'Transportes Medicos' presentó un informe de requerimientos en los que solicitaba ciertos cambios, todos los cuales eran de mantenimiento y no de estructura, es decir no obligaban a un rediseño de la aplicación, en su mayoría eran la manera como se presentaban algunos reportes, simplemente se tenía que presentar la misma información en otro orden.

Tres meses después que la aplicación había estado corriendo en la red, se vio que la velocidad de procesamiento se había degradado en un porcentaje que aunque no representaba mayor desventaja si podía tornarse preocupante si no se corregía a tiempo (la cantidad de pacientes que se transportaban diariamente

habian sobrepasado las estimaciones iniciales). por lo tanto se revisó la aplicación en puntos que podían ser críticos concernientes a la velocidad de procesamiento y se rediseñaron las rutinas para ser mas eficientes.

Es interesante notar que para el mantenimiento de la aplicacion, esta se facilitaba debido a que se habia utilizados programacion estructurada en su elaboracion, y en la medida de lo posible se habia evitado el tener programas de gran longitud. Asi mismo en la medida de lo posible se habia tratado de uniformar el estilo de programacion, en el anexo al final del informe se muestra el codigo fuente.

9.- Anexos

9.1 Listado de los programas

9.2 Ejemplo de factura emitidas por la aplicacion

9.3 Tarjeta ó Ticket del paciente

9.4 Formato de requerimiento de pagos (Medicaid)


```

# -----
# PROGRAMA : MAIN
# OBJETIVO : MUESTRA pantalla general del menu de Accounting
# AUTOR    : JUAN M. CHAVEZ
# DATE     : Septiembre 1990
# -----
clear all
#set compatible db4
#on error do acerror with program(), lineno(), error()
do def_amb

close all
close data
public en_name, en_addr, en_city, en_sta, en_zip, en_ph, en_fax, utot_in, vclase

en_name = 'AMT MEDICAL TRANSP.'
en_addr = 'P.O. Box 2068'
en_city = 'Miami'
en_sta  = 'F1'
en_zip  = '33152'
en_ph   = '(305) 529-9433'
en_fax  = '(305) 599-9413'
en_hrs  = '089271800'
vclase  = 'S'
utot_in = 0

wenter = chr(17) + chr(196) + chr(217)
wespr  = 0
clave  = 'master'
# llama clave de seguridad
# -----

set typeahead to 10

if iscolor()
# Tomar valores de cod_seq de campo color_mo de cod_seq.dbf
# popup color
color_mo = 1
do case
case color_mo = 1
set colo to gr+/b,u+/r,n
# menu principal
set color of scheme 1 to gr+/b,u+/r,,u+/r,r+/b,r+/u,gr+/n,n+
set color of scheme 2 to n/bg,u+/bg,n/bg,u/r,r+/b,u+/r
# 1er popup
set color of scheme 17 to u+/rb,u+/rb,u+/rb,u+/rb,r+/b,u+/r,b/rb,n+
# 2do popup
set color of scheme 18 to u+/bg,n/bg,u+/bg,,r+/b,u+/r,,n+
# 3er popup
set color of scheme 23 to u+/bg,u+/r,r/u,n/u,b+/b,gr/n,b/bg

# 1ra ventana
set color of scheme 19 to b/bg,n/u,n/bg
# 2da ventana
set color of scheme 20 to n/bg,n/u,n/bg,,r+/b,u+/r

```

```

* 3ra ventana
set color of scheme 22 to n/g,n/v,n/g,,r+/b,w+/r
set color of scheme 3 to n/g,n/g,n/g,,r+/b,w+/r
* Se utiliza para el menu principal
set color of scheme 24 to gr+/b,gr+/b,bg/v,r+/b,r+/b,w+/r
set color of scheme 21 to gr+/b,gr+/b,bg/v,r+/b,r+/b,w+/r

case color_mo = 2
set colo to br+/v,bg+/n,w+/v
* menu principal
set color of scheme 1 to n/v,n/bg,,n/bg,w+/v,r+/v,gr+/n,nt
set color of scheme 2 to n/bg,w+/bg,n/bg,v/r,w+/v,w+/r
* 1er popup
set color of scheme 17 to b+/b,bg+/b,w+/b,,w+/v,n/bg,,n
* 2do popup
set color of scheme 18 to ,w+/gr,w+/gr,,w+/v,n/bg,,nt
* 1ra ventana
set color of scheme 19 to w+/r,b+/v,w+/r
* 2da ventana
set color of scheme 20 to n/bg,w+/r,n/bg,,r+/b,w+/r
* otra ventana
set color of scheme 21 to gr+/b,gr+/b,bg/v,r+/b,r+/b,w+/r
* 3ra ventana
set color of scheme 22 to n/g,n/v,n/g,,r+/b,w+/r
* 3er popup
set color of scheme 23 to w+/rb,w+/rb,w+/rb,,w+/v,w+/bg,,nt
* Se utiliza para el menu principal
set color of scheme 24 to gr+/v,g+/v,bg/v,w+/v,w+/v,w+/r

case color_mo = 3
set colo to br+/n,bg+/n,n
* menu principal
set color of scheme 1 to gr+/n,n/bg,,n/bg,w+/n,r+/v,gr+/n,nt
set color of scheme 2 to n/bg,w+/bg,n/bg,v/r,w+/v,w+/r
* 1er popup
set color of scheme 17 to w+/r,w+/r,w+/r,,w+/n,w+/rb,,n
* 2do popup
set color of scheme 18 to ,n/v,w+/v,,w+/n,w+/rb,,nt
* 1ra ventana
set color of scheme 19 to b/v,w+/r,n/v
* 2da ventana
set color of scheme 20 to w+/b,w+/r,w+/b,,w+/n,w+/r
* otra ventana
set color of scheme 21 to gr+/b,gr+/b,bg/v,r+/b,w+/n,w+/r
* 3ra ventana
set color of scheme 22 to w+/b,n/v,w+/b,,w+/n,g+/n
* 3er popup
set color of scheme 23 to w+/rg,w+/rg,w+/rg,,w+/n,w+/g,,nt
* Se utiliza para el menu principal
set color of scheme 24 to gr+/v,r+/n,bg/v,w+/v,w+/n,w+/r

case color_mo = 4
set colo to br+/n,bg+/n,n
* menu principal
set color of scheme 1 to w+/bg,w+/b,,n/bg,rg+/bg,w+/r,gr+/n,nt
set color of scheme 2 to n/bg,w+/bg,n/bg,v/r,w+/v,w+/r

```

```

* 1er popup
set color of scheme 17 to v+/g,v+/g,n+/g,,rg+/bg,v+/r,,n
* 2do popup
set color of scheme 18 to    ,n/v,v+/v,,rg+/bg,v+/r,,n+
* 1ra ventana
set color of scheme 19 to n/v,v+/r,gr+/v
* 2da ventana
set color of scheme 20 to v+/b,v+/r,v+/b,,rg+/bg,v+/r
* otra ventana
set color of scheme 21 to gr+/b,gr+/b,bg/v,r+/b,rg+/bg,v+/r
* 3ra ventana
set color of scheme 22 to v+/b,n/v,v+/b,,rg+/bg,g+/n
* 3er popup
set color of scheme 23 to  v+/r,v+/r,v+/r,,rg+/bg,n/v,,n+
* Se utiliza para el menu principal
set color of scheme 24 to gr+/v,n/bg,bg/v,v+/v,rg+/bg,v+/r

case color_mo = 5
set colo to br+/n,bg+/n,n
* menu principal
set color of scheme 1  to v+/br,v+/b,,n/bg,rg+/br,bg+/v,gr+/n,n+
set color of scheme 2  to n/bg,v+/bg,n/bg,v/r,v+/v,v+/r
* 1er popup
set color of scheme 17 to n/v,n/v,n/v,,rg+/br,v+/r,,n
* 2do popup
set color of scheme 18 to    ,n/g,v+/g,,rg+/br,v+/b,,n+
* 1ra ventana
set color of scheme 19 to n/v,v+/r,gr+/v
* 2da ventana
set color of scheme 20 to n/bg,n/v,n/bg,,rg+/br,v+/r
* otra ventana
set color of scheme 21 to gr+/b,gr+/b,bg/v,r+/b,rg+/br,v+/r
* 3ra ventana
set color of scheme 22 to v+/b,n/v,v+/b,,rg+/br,g+/n
* 3er popup
set color of scheme 23 to  v+/bg,n/bg,n/bg,,rg+/br,v+/b,,n+
* Se utiliza para el menu principal
set color of scheme 24 to gr+/v,gb+/br,bg/v,v+/v,gr+/br,v+/r

case color_mo = 6
* popup monocromatico
* un solo color
set color of scheme 1  to v/n,v+/n,v/n,v/n,n/v,v/n
set color of scheme 2  to v/n,v+/n,n+/v,v/n,v/n,n+/v
set color of scheme 17 to v/n,v+/n,v/n,n/v,n/v,n/v
set color of scheme 18 to v/n,v+/n,v/n,n/v,n/v,n+/v

set color of scheme 19 to v/n,n/v,v/n,n/v,n/v,n/v
set color of scheme 20 to v/n,n/v,n/v,n/v,n/v,n+/v
set color of scheme 21 to v/n,n/v,n/v,n/v,n/v,n/v
set color of scheme 22 to v/n,n/v,n/v,n/v,n/v,n+/v
set color of scheme 23 to v/n,v+/n,v/n,n/v,n/v,n+/v
* solo menu princ:
set color of scheme 24 to v/n,v/n,v/n,n/v,n/v,n/v
set shadows off
set colo to ,n/v,n

```

```

    endcase
else
    * popup monocromatico
    * un solo color
    set color of scheme 1 to ,n/v,n/v,v/n,v/n,v/n
    set color of scheme 2 to ,v+/n,n+/v,v/n,v/n,n+/v
    set color of scheme 17 to ,v+/n,v/n,n/v,n/v,n/v
    set color of scheme 18 to ,v+/n,v/n,n/v,n/v,n+/v

    set color of scheme 19 to ,n/v,v/n,n/v,n/v,n/v
    set color of scheme 20 to ,n/v,n/v,n/v,n/v,n+/v
    set color of scheme 21 to ,n/v,n/v,n/v,n/v,n/v
    set color of scheme 22 to ,n/v,n/v,n/v,n/v,n+/v
    set color of scheme 23 to ,n/v,n/v,v/n,v/n,n+/v,v/n,v/n
    * solo menu princ:
    set color of scheme 24 to ,v/n,v/n,n/v,n/v,n/v
    set colo to v/n,n/v,n,v/,v/n,v/n,v/n,v/n,v/n,v/n
    set shadows off

endif

define windo inicio from 0,0 to 23,79 none color scheme 1
activa windo inicio
set stat on
clear

close data

@ 0,0 say em_name
@ 1,0 say em_addr
@ 2,0 say em_city + ', ' + em_sta + ' ' + em_zip
* @ 20,35 say date()
@ 3,0 to 3,79 double
@ 21,0 to 23,79
@ 22,3 say 'Use arrows to select ; ' + venter + ' to pickup selection ; ESC to cancel'

define popup factura from 6,22 to 17,54 shadow title ' MAIN MENU ' color scheme 17
define bar 1 of factura prompt ' A. Ticket trips' message 'To Create, modify, delete ticket trips'
define bar 2 of factura prompt ' B. Billing' message 'To process the ticket trip'
define bar 3 of factura prompt ' C. Customers' messa 'To create new companies, delete, update'
define bar 4 of factura prompt ' D. Receivables' message 'To register payments for the invoices.'
define bar 5 of factura prompt ' E. Utilities' message 'Create, modify, delete operators, check files, new rates, etc.'
define bar 6 of factura prompt ' F. CAC-RAMSAY Claims' message 'Billing the tickets trip to CAC-RAMSAY'
define bar 7 of factura prompt ' ' skip
define bar 8 of factura prompt ' 6. Exit' message 'To finish y return to DOS'

on selection popup factura do facturas

activate popup factura
deactivate popup
close databases
release popups factura
deacti windo all
set stat off

```

```

set colo to
set talk on
set debug on
close databa
clear

return

# -----
procedure facturas

# Parametros que abarcan a todas las opciones
do case
case bar() = 1
  do tickets
case bar() = 2
  do menubill
case bar() = 3
  do accounts
case bar() = 4
  do paymenu
case bar() = 5
  do params
case bar() = 6
  do bill_cac
case bar() = 8

  do fin
endcase

# -----
return
#-----
# PROGRAMA FIN DE SESION
#-----
procedure fin
define windo adios from 13,10 to 17,68 shadow color scheme 19
activa windo adios
@ 0,0 say "      Software developed by Technotel, Inc."
@ 1,0 say '          1992 - (305) 279-3861'

@ 2,0 say '          Press any key'
wait ''
deactiva windo adios
clear
deactivate menu
set talk on
set statu off
set debug on
on error
deacti windo all
cancel
return
#quit
# -----

```

```
procedure def_amb
# define el ambiente de trabajo
set exclusive off
set date to american
set escape off
set score off
set border to single
set talk off
set typeahead to 0
set near on
set dele on
set bell off
set exact on
set print off
set reprocess to automatic
set devi to scree
set bell off
set century off
set stat off
set sysmenu off
set shadows on
on key label f1 dq2 = ''
on key label f2 dq2 = ''
on key label f3 dq2 = ''
on key label f5 dq2 = ''
on key label f6 dq2 = ''
on key label f7 dq2 = ''
on key label f8 dq2 = ''
on key label f10 dq2 = ''
return
```

```

-----
PROGRAMA : acfactu
OBJETIVO : MUESTRA pantalla general del menu de Facturacion
AUTOR    : JUAN M. CHAVEZ
FECHA    : Septiembre 1990
-----

```

DEFINE MENU POPUP DE FACTURAS

```

define popup tiket from 11,40 to 20,68 shadow color scheme 18
define bar 1 of tiket prompt ' A. Create Tickets' message 'To Create Tickets'
define bar 2 of tiket prompt ' B. Modify Tickets' message 'To change tickets'
define bar 3 of tiket prompt ' C. Print Tickets' message 'To print Tickets'
define bar 4 of tiket prompt ' D. Delete tickets' message 'Delete Tickets'
define bar 5 of tiket prompt ' E. Daily Route Sheet' message 'Create Daily Route Sheet'
define bar 6 of tiket prompt ' F. Tickets report' message 'Daily report of tickets'
define bar 7 of tiket prompt ' G. Regular Patients file' message 'To enter, modify, delete regular patients'

```

```

include popup tiket do tickets

```

```

define popup tiket
define popup
define databases
define popups tiket
end

```

define tickets

```

define metros que abarcan a todas las opciones
define se
define bar() = 1
define crea_ti
define bar() = 2
define moditi
define bar() = 3
define printick
define bar() = 4
define deleti
define bar() = 5
define day_rout
define bar() = 6
define ti_repo
define bar() = 7
define regupa
define DO acesti
define ase
define databases
end

```

```

* -----
* PROGRAMA : menubill
* OBJETIVO : MUESTRA pantalla general del menu de Facturacion
* AUTOR   : JUAN M. CHAVEZ
* DATE    : Septiembre 1990
* -----
*-----
* DEFINE MENU POPUP DE FACTURAS
*-----
define popup bill from 12,10 to 19,40 shadow color scheme 18
define bar 1 of bill prompt ' A. Check Invoices' message 'To enter mileage, additional items for invoicing'
define bar 2 of bill prompt ' B. Print Invoices' message 'To generate invoices: Printing, and Medicaid forms'
define bar 3 of bill prompt ' C. Print Medicaid'  message 'To print Medicaid forms (HRS), by date'
define bar 4 of bill prompt ' D. Delete Invoice '  message 'To delete invoices'
define bar 5 of bill prompt ' E. Summary Report' message 'Print total invoicing for given dates.'
*DEFINE BAR 6 OF bill PROMPT ' F. Tickets report' MESSAGE 'Daily report of tickets'
*define bar 7 of bill prompt '\-----'
*DEFINE BAR 8 OF bill PROMPT ' 6. Estimados/Cotizaciones' MESSAGE 'Consultar, crear, imprimir, modificar estimados'

on selection popup bill do bills

activate popup bill
*DEACTIVATE POPUP
close databases
*release popups bill
return

* -----
procedure bills

* Parametros que abarcan a todas las opciones
do case
case bar() = 1
do chk_invo
case bar() = 2
do pr_invo
case bar() = 3
do pr_caaid
case bar() = 4
do delein
case bar() = 5
do prsales

endcase

close databases

return

```



```

* -----
* PROGRAMA : paymenu
* OBJETIVO : MUESTRA pantalla general del menu de Payables
* AUTOR    : JUAN M. CHAVEZ
* DATE     : April 1992
* -----
*-----
* DEFINE MENU POPUP DE FACTURAS
*-----
define popup payab from 13,10 to 19,40 shadow color scheme 18
define bar 1 of payab prompt ' A. Check Payments' message 'To register payments received from customers'
define bar 2 of payab prompt ' B. Outstanding Invoices' message 'Report of invoices that have not been cancelled'
define bar 3 of payab prompt ' C. Summary of payments'   message 'Report of cancelled invoices.'
define bar 4 of payab prompt ' D. Total outstanding'     message 'Report of total dues by company'

on selection popup payab do payabs

activate popup payab
*DEACTIVATE POPUP
close databases
*release popups payab
return

* -----
procedure payabs

* selecciona areas de trabajo
* -----
select 1
use tickets index tickets order ticket_nbr
select 3
use paramet
select 2
use accounts index accounts order acct_nbr

* Parametros que abarcan a todas las opciones
do case
case bar() = 1
  do chk_pay
case bar() = 2
  do pr_pay
case bar() = 3
  do pr_cancel
case bar() = 4
  do pr_dues
case bar() = 5
  * do prsales

endcase

close databases
return

*****

```

```

* end main program
* *****
procedure pr_pay
define windo prcust from 6,5 to 22,75 shadow color scheme 19
activa windo prcust
@ 0,0 say 'Report of outstanding invoices'
@ 1,0 say '-----'
@ 2,0 say 'You will have a list of invoices billed but not yet canceled'
vaccount = space(6)
vfeini = date()
wfefin = date()
vdevi = 'P'
set messa to 'Press ESC to return. ' + venter + ' next field'
@ 4,0 say 'Acct. #:' get vaccount picture 'xxxxxx' valid lista(vaccount)
@ 5,0 say '(Type ***** if you want all of them)'

@ 7,0 say 'Billed from ' get vfeini
@ 7,24 say 'To ' get wfefin
@ 9,0 say 'To Printer/Screen ? (P/S) : ' get vdevi picture '@M P,S'
read
if lastkey() = 27
  deacti windo prcust
  return
endif
set messa to 'Printing .....wait'

select tickets

do case
case vaccount = '*****'
  set order to acct_nbr
  if vdevi = 'P'
    report form py_acct for date_issue >= vfeini .and. date_issue <= wfefin ;
    .and. posted = 'Y'.and. tot_invo - pymt <> 0 heading '***** OUTSTANDING INVOICES *****' noeject noconsole to print
  else
    report form py_acct for date_issue >= vfeini .and. date_issue <= wfefin ;
    .and. posted = 'Y' .and. alltrim(acct_nbr) = alltrim(vaccount) ;
    .and. tot_invo - pymt <> 0 heading '***** OUTSTANDING INVOICES *****' preview
  endif
otherwise
  set order to billed
  if vdevi = 'P'
    report form py_acct for date_issue >= vfeini .and. date_issue <= wfefin ;
    .and. posted = 'Y' .and. alltrim(acct_nbr) = alltrim(vaccount) ;
    .and. tot_invo - pymt <> 0 heading '***** OUTSTANDING INVOICES *****' noeject noconsole to printer
  else
    report form py_acct for date_issue >= vfeini .and. date_issue <= wfefin ;
    .and. posted = 'Y' .and. alltrim(acct_nbr) = alltrim(vaccount) ;
    .and. tot_invo - pymt <> 0 heading '***** OUTSTANDING INVOICES *****' preview
  endif
endcase

```

```
deactiva windo prcust
return
```

```
procedure pr_cancel
define windo prcust from 6,5 to 22,75 shadow color scheme 19
activa windo prcust
@ 0,0 say 'Report of cashed invoices'
@ 1,0 say '-----'
@ 2,0 say 'You will have a list of invoices billed and cashed'
vaccount = space(6)
vfeini = date()
vfein = date()
vdevi = 'P'
set messa to 'Press ESC to return.' + wenter + ' next field'
@ 4,0 say 'Acct. #:' get vaccount picture 'xxxxxx' valid lista(vaccount)
@ 5,0 say '(Type ***** if you want all of them)'
```

```
@ 7,0 say 'Billed from ' get vfeini
@ 7,24 say 'To ' get vfein
@ 9,0 say 'To Printer/Screen ? (P/S) : ' get vdevi picture 'EM P,S'
read
if lastkey() = 27
  deacti windo prcust
  return
endif
set messa to 'Printing .....wait'
```

```
select tickets
```

```
do case
case vaccount = '*****'
  set order to acct_nbr
  if vdevi = 'P'
    report form py_acct for date_issue >= vfeini .and. date_issue <= vfein ;
    .and. posted = 'Y'.and. tot_invo - pynt = 0 heading '*** P A I D   I N V O I C E S ***' noeject noconsole to print
  else
    report form py_acct for date_issue >= vfeini .and. date_issue <= vfein ;
    .and. posted = 'Y' .and. tot_invo - pynt = 0 heading '*** P A I D   I N V O I C E S ***' preview
  endif
endif
```

```
otherwise
  set order to billed
  if vdevi = 'P'
    report form py_acct for date_issue >= vfeini .and. date_issue <= vfein ;
    .and. posted = 'Y' .and. alltrim(acct_nbr) = alltrim(vaccount) ;
    .and. tot_invo - pynt = 0 heading '*** P A I D   I N V O I C E S ***' noeject noconsole to printer
  else
    report form py_acct for date_issue >= vfeini .and. date_issue <= vfein ;
    .and. posted = 'Y' .and. alltrim(acct_nbr) = alltrim(vaccount) ;
    .and. tot_invo - pynt = 0 heading '*** P A I D   I N V O I C E S ***' preview
  endif
endif
```

```
endcase
```

```
deactiva vindo prcust
return
```

```
function lista
parameters vcode
if alltrim(vcode) = '*****'
  return .t.
endif
seek trim(vcode)
if .not. found()
  select accounts
  set order to company
  define popup taller1 from 0,32 to 14,68 prompt field company ;
  message 'Arrows to move, or first letter. ' + venter + ' to select' shadow color scheme 17
  on selection popup taller1 do ruteata with prompt()
  activate popup taller1
  vaccount = accounts->acct_nbr
endif
```

```
@ 4,19 say accounts->company picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
set order to acct_nbr
return .t.
```

```
procedure ruteata
parameters empresa
valor = 0
```

```
seek empresa
if .not. found()
  return
else
  valor = 1
  if lastkey() = 13
    deacti popup
    vaccount = accounts->acct_nbr
  endif
endif
return
```

```
*****
procedure pr_dues
define vindo prcust from 6,5 to 19,75 shadow color scheme 19
activa vindo prcust
@ 0,0 say 'Report of total dues by company'
@ 1,0 say '-----'
@ 2,0 say 'You will have a list of totals billed but not yet canceled'
vaccount = space(6)
vfeini = date()
vfein = date()
vdevi = 'P'
set messa to 'Press ESC to return. ' + venter + ' next field'
vaccount = '*****'
```

```
@ 5,0 say 'Billed from ' get vfeini
@ 5,24 say 'To ' get vfein
@ 8,0 say 'To Printer/Screen ? (P/S) : ' get vdevi picture '@M P,S'
read
```

```

if lastkey() = 27
  deacti windo prcust
  return
endif
set messa to 'Printing .....wait'

select tickets

do case
case vaccount = '*****'
  set order to acct_nbr
  if wdevi = 'P'
    report form py_acct1 for date_issue >= vfeini .and. date_issue <= wfefin ;
      .and. posted = 'Y'.and. tot_invo - pymt <> 0 heading '***** OUTSTANDING INVOICES *****' noobject noconsole to printe
  else
    report form py_acct1 for date_issue >= vfeini .and. date_issue <= wfefin ;
      .and. posted = 'Y' .and. tot_invo - pymt <> 0 heading '***** OUTSTANDING INVOICES *****' preview
  endif
endif

endcase

deactiva windo prcust
return

```

```

* -----
* PROGRAMA : params
* OBJETIVO : MUESTRA pantalla general del menu de Parametros
* AUTOR    : JUAN M. CHAVEZ
* DATE     : Juli 1993
* -----
* ----
* DEFINE MENU POPUP DE PARAMETROS
* -----
define popup param from 11,45 to 15,70 shadow color scheme 18
define bar 1 of param prompt ' A. Extract old Tickets' message 'Extract old Tickets and Invoices'
define bar 2 of param prompt ' B. Organize Invoices' message 'Reorganize Invoice file'

```

```

on selection popup param do paramt

```

```

activate popup param
close databases
return

```

```

* -----
procedure paramt

```

```

* Parametros que abarcan a todas las opciones
do case
case bar() = 1
  do utilit
case bar() = 2
  * DO packinvo

```

```

* CASE BAR() = 3
* DO bill_cac
endcase

```

```

close databases
return

```

```

# -----
# PROGRAMA : Bill-cac: OBJETIVO : MUESTRA pantalla general del menu de Billi using magnetic device

```

```

# AUTOR   : JUAN M. CHAVEZ
# DATE    : Juli 1993
# -----
# ---

```

```

# DEFINE MENU POPUP DE BILL_cac
# -----

```

```

define popup billc from 13,37 to 18,73 shadow color scheme 18
define bar 1 of billc prompt ' A. Prepare Billing claims' message 'To select claims by Invoices or by range-date'
define bar 2 of billc prompt ' B. Create Billing file' message 'Create Bill-file to be send to CAC-RANSAY (ASCII)'
define bar 3 of billc prompt ' C. Reports' message 'Several Reports'

```

```

on selection popup billc do billingc

```

```

activate popup billc
close databases
return

```

```

# -----
procedure billingc

```

```

# Parametros que abarcan a todas las opciones
do case
case bar() = 1
do bill_sel
case bar() = 2
do bill_cre
case bar() = 3
do repo_cac
endcase

```

```

close databases
return

```

```

# menu for reports

```

```

procedure repo_cac

```

```

define popup bille from 17,8 to 22,41 shadow color scheme 3

```

```

define bar 1 of bille prompt ' A. Claims billed to CAC-RANSAY' message 'Invoices Billed to CAC-RANSAY'
define bar 2 of bille prompt ' B. Claims ready to send' message 'Print out of complete claims to be transmitted by modem'
define bar 3 of bille prompt ' C. Uncomplete claims' message 'Claims not completed (without author. #, Patient #, etc.)'

```

```

* crea tickets
* -----

define window panta from 0,0 to 23,79 none
activa wind panta

* define popups
* -----

* selecciona areas de trabajo
* -----
select l4
use regu_ti index regu_ti order patient_nu

select l
use tickets index tickets order ticket_nbr
select 3
use paramet
select 2
use accounts index accounts order acct_nbr

* inicializa variables
* -----
vticket = 0
vaccount = space(5)
vissue = date()
vcall_by = space(20)
vph_caller = space(14)
vhora = 0
vmin = 0
vam_pm = 'P'
vcall_taker = space(15)
vlast_name = space(40)
vfirst = space(40)

vfrom_add = space(45)
vfrom_ci = 'Miami' + space(20)
vfrom_sta = 'F1'
vfrom_zip = space(10)

vto_add = space(45)
vto_ci = 'Miami' + space(20)
vto_sta = 'F1'
vto_zip = space(10)

voat_phone = space(14)
vtrip = '!_Way'
whurry = 'Rout.'
vbill_to = space(40)
vbill_add = space(45)
vbill_ci = 'Miami' + space(20)
vbill_sta = 'F1'
vbill_zip = space(10)

vcare = space(18)
vcaid = space(22)
vpat_nbr = space(22)

```

```

vpat_member = space(22)
vauthori = space(20)
vtype_svc = 'Amb.'

vsend_add = space(45)
vsend_ci = 'Miami' + space(20)
vsend_sta = 'F1'
vsend_zip = space(10)
vposted = 'N'

vcoment1 = space(40)
vcoment2 = space(40)

vsw_regu = 0

@ 0,0 say 'New Ticket Trip: '
@ 1,0 to 1,79 double
@ 23,0 to 23,79 double
set messa to 'Press F4 to get regular patients. ESC to cancel. '+ venter + ' next field'

* activa function key para pacientes regulare

on key label f4 do listait

@ 6,0 to 16,79
@ 6, 32 say ' Patient: '

do while .t.

  * muestra en pantalla
  * -----
  @ 0,60 say 'Serv. date'
  @ 0,71 get wissue
  @ 2,0 say 'Acct. #:' get waccount picture 'xxxxxx' valid lista(waccount)

  @ 2,35 say 'Bill:' get wbill_to picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
  @ 3,35 say 'Addr:' get wbill_add picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
  @ 4,35 say 'City:' get wbill_ci picture 'XXXXXXXXXXXXXXXXXXXX'
  @ 4,60 say 'St:' get wbill_sta picture 'XX'
  @ 4,69 say 'Zip:' get wbill_zip picture 'XXXXXX'

  @ 3,0 say 'Call by:' get wcall_by picture 'XXXXXXXXXXXXXXXXXXXX'
  @ 4,0 say 'Phone #' get wph_caller picture 'XXXXXXXXXXXXXXXXXX'

  @ 7,1 say 'Last Name:' get wlast_name picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
  @ 8,1 say 'First Name:' get wfirst picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
  @ 9,1 say 'Pat. phone:' get wpat_phone picture 'XXXXXXXXXXXXXXXXXX'

  @ 7,54 say 'Caid:' get wcaid picture 'XXXXXXXXXXXXXXXXXXXX'
  @ 8,50 say 'Member #:' get vpat_member picture 'XXXXXXXXXXXXXXXXXXXX'
  @ 9,50 say 'Pat.Nbr.:' get vpat_nbr picture 'XXXXXXXXXXXXXXXXXXXX'
  @ 10,50 say 'Auth. #:' get vauthori picture 'XXXXXXXXXXXXXXXXXXXX'
  @ 11,50 say 'Medicare:' get vcare picture 'XXXXXXXXXXXXXXXXXXXX'

  @ 11,1 say 'From:' get wfrom_add picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'

```



```

@ !2,1 say 'City:' get wfrom_ci picture 'XXXXXXXXXXXXXXXXXXXXX'
@ !2,29 say 'State:' get wfrom_sta picture 'XX'
@ !2,41 say 'Zip:' get wfrom_zip picture 'XXXXXXXXXX'

@ 14,1 say 'To : ' get wto_add picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ 15,1 say 'City:' get wto_ci picture 'XXXXXXXXXXXXXXXXXXXXX'
@ 15,28 say 'State:' get wto_sta picture 'XX'
@ 15,41 say 'Zip:' get wto_zip picture 'XXXXXXXXXX'

@ 14,62 say 'Time P/U Req.'
@ 15,62 get whora picture '99' range 0,12
@ 15,65 say ':'
@ 15,67 get vmin picture '99' range 0,60
@ 15,73 say 'Am/Pm'
@ 15,71 get wam_pm picture '@M A,P'

@ 17,0 say 'Call Taker:' get wcall_taker picture 'XXXXXXXXXXXXXXXXXXXXX'

@ 19,0 say '1 Way/Round Trip:' get wtrip picture '@M 1_Way,RTrip'
@ 19,29 say 'Amb./WheelCh./Str.:' get wtype_svc picture '@M Amb.,W/C,Str.'
@ 19,59 say 'Routine/ASAP:' get whurry picture '@M Rout.,ASAP'

@ 21,0 say 'Notes:' get wcoment1 picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ 22,7 get wcoment2 picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
read

if lastkey() = 27
  deacti windo panta
  on key label f4 ver = ''
  return
endif

* chequea campos
* -----
if len(alltrim(wbill_add)) < 2 .or. len(alltrim(wbill_ci)) < 1 .or. ;
  len(alltrim(wbill_sta)) < 1 .or. len(alltrim(wbill_zip)) < 0
  set messa to 'Error: Check Address "Bill to:" '
  loop
endif

if len(alltrim(wbill_to)) < 2
  set messa to 'Error: Check "Bill to:" '
  loop
endif

if len(alltrim(wlast_name)) < 2 .or. len(alltrim(wfirst)) < 1
  * len(alltrim(wpat_phone)) < 1
  set messa to "Error: Check Patient's name "
  loop
endif

if len(alltrim(wfrom_add)) < 2 .or. len(alltrim(wfrom_ci)) < 1 .or. ;
  len(alltrim(wfrom_sta)) < 1 .or. len(alltrim(wfrom_zip)) < 0
  set messa to "Error: Check patient's address 'From:' "
  loop
endif

```

```

if len(alltrim(wto_add)) < 2 .or. len(alltrim(wto_ci)) < 1.or. ;
  len(alltrim(wto_sta)) < 1 .or. len(alltrim(wto_zip)) < 0
  set messa to "Error: Check patient's address 'To:'"
  loop
endif

if !len(alltrim(wcall_taker)) < 2
  set messa to 'Error: Who is the call taker ?'
  loop
endif

set messa to 'Press ESC to cancel'
* menu para preguntar al finalizar ticket
* -----
define windo vmensa from 16,10 to 20,72 shadow color scheme 19
wrpta = 'G'
define menu opci_fi
define pad grabaim of opci_fi prompt "Save and Print" at 1,1
define pad graba of opci_fi prompt "Save Only" at 1,20
define pad cancela of opci_fi prompt "Cancel" at 1,36
define pad revisa of opci_fi prompt "Check Again" at 1,48
define pad acctnu of opci_fi prompt "Save, Print & create acct. nbr." at 2,14

on selection pad grabaim of opci_fi do final with wrpta
on selection pad graba of opci_fi do final with wrpta
on selection pad cancela of opci_fi do final with wrpta
on selection pad revisa of opci_fi do final with wrpta
on selection pad acctnu of opci_fi do final with wrpta

* Pregunta para confirmar si graba ticket
* -----
define window vpre from 16,5 to 21,72 shadow color scheme 20

activa windo vpre
@ 0,1 say 'Ticket ready:'
wrpta = 'R'

do while .t.
  set messa to 'Use arrows to highlight (-, -) . ' + venter + ' to select'
  activa menu opci_fi

  if !astkey() = 27
    deact wind vpre
    set messa to
    activa windo vmensa
    wrpta = 'N'
    @ 1,0 say 'Do you want to return without saving ticket ? (Y/N -> ' get wrpta picture ''
    read
    deactiva windo vmensa
    if wrpta = 'Y'

      deactiva windo panta

    return
  
```

```

    else
        acti wind vpre
        @ 0,1 say 'Ticket ready:'
        loop
    endif

endif

exit
enddo

deactiva windo vpre
set messa to
do case
case vrpta = 'G'
do graba_ti
do tick_pr with ticket_nbr
case vrpta = 'S'
do graba_ti
case vrpta = 'C'
deactiva windo panta
return
case vrpta = 'A'
do graba_ti
wnro_acct = 0
if vsw_regu = 0
* solamente genera una cuenta cuando la informac. es nueva
do gene_act with wnro_acct
endif
do tick_pr with ticket_nbr
otherwise
loop
endcase

activa windo vmensa
set messa to
if vrpta = 'A'
if vsw_regu = 0
@ 0,15 say 'Acct. nbr. for this patient: ' + alltrim(str(wnro_acct))
else
@ 0,18 say 'Acct. already created'
endif
endif
if vsw_regu = 1
select tickets
replace tickets->nro_regupa with regu_ti->patient_nu
endif
@ 1,18 say '**** Ticket created ****'
wait '          Press any key for the next one'
deacti windo vmensa

* reinicializa variables para siguiente ticket
* -----
vticket = 0
vaccount = space(5)
vissue = date()
vcall_by = space(20)

```

```

wph_caller = space(14)
whora = 0
wmin = 0
wam_pm = 'P'
wvcall_taker = space(15)
wlast_name = space(40)
wfirst = space(40)

vfrom_add = space(45)
vfrom_ci = 'Miami' + space(20)
vfrom_sta = 'F1'
vfrom_zip = space(10)

vto_add = space(45)
vto_ci = 'Miami' + space(20)
vto_sta = 'F1'
vto_zip = space(10)

vpat_phone = space(14)
vtrip = '1_Way'
vhurry = 'Rout.'

vbill_to = space(40)
vbill_add = space(45)
vbill_ci = 'Miami' + space(20)
vbill_sta = 'F1'
vbill_zip = space(10)

vcare = space(18)
vcaid = space(22)
vpat_nbr = space(22)
vpat_member = space(22)
vauthori = space(20)
vtype_svc = 'Amb.'

vsend_add = space(45)
vsend_ci = 'Miami' + space(20)
vsend_sta = 'F1'
vsend_zip = space(10)
vposted = 'N'

vcomment1 = space(40)
vcomment2 = space(40)

vsw_regu = 0

@ 6,0 to 16,79
@ 6, 32 say ' Patient: '
@ 0,0 say 'New Ticket Trip: ' + space(20)
@ 2,15 say space(25)
set messa to 'Press ESC to cancel Data Entry. '+ venter + ' next field'
select accounts

enddo

return

```

```

*****
* SUBROUTINES - END OF MAIN PROGRAM
* -----

function lista
parameters wcode
select accounts
set order to acct_nbr
seek trim(wcode)
if .not. found()
  select accounts
  set order to company
  define popup taller! from 3,32 to 20,70 prompt field company ;
  message 'Arrows to move, or first letter., ' + wenter + ' to select' shadow color scheme 17
  on selection popup taller! do ruteata with prompt()
  activate popup taller!
  vaccount = accounts->acct_nbr
endif

vbill_to = accounts->company
vbill_add = accounts->address
vbill_ci = accounts->city
vbill_sta = accounts->state
vbill_zip = accounts->zip
vaccount = accounts->acct_nbr

show get vaccount
show get vbill_to
show get vbill_ci
show get vbill_sta
show get vbill_zip
show get vbill_add

@ 2,15 say accounts->company picture 'XXXXXXXXXXXXXXXXXX'

set order to acct_nbr
return .t.

procedure ruteata
parameters empresa
valor = 0

seek empresa
if .not. found()
  return
else
  valor = 1
  if lastkey() = 13
    deacti popup
    vaccount = accounts->acct_nbr
    show get vaccount
  endif
endif
return

procedure final

```

```

parameters wrpta
* para saber que opcion se ha elegido
do case
case pad() = 'GRABAIN'
  wrpta = 'G'
case pad() = 'GRABA'
  wrpta = 'S'
case pad() = 'CANCELA'
  wrpta = 'C'
case pad() = 'REVISA'
  wrpta = 'R'
case pad() = 'ACCTNU'
  wrpta = 'A'

endcase
deacti menu
return

procedure graba_ti
set messa to 'Please wait..... saving ticket'
select paramet
if flock()
  vticket = paramet->nxt_ticket
  replace nxt_ticket with vticket + 1

  select tickets
  append blank
  replace ticket_nbr with vticket, acct_nbr with vaccount, account with accounts->company, date_issue with vissue, ;
  call_by with vcall_by, ph_caller with vph_caller, hor_pickup with vhora, ;
  min_pickup with vmin, am_pm with vam_pm, call_taker with vcall_taker, ;
  last_name with vlast_name, first_name with vfirst, from_addr with vfrom_add

  replace from_city with vfrom_ci, from_state with vfrom_sta, from_zip with vfrom_zip, bill_to with vbill_to, ;
  to_addr with vto_add, to_city with vto_ci, to_state with vto_sta, to_zip with vto_zip, pat_phone with vpat_phone, ;
  trip with substr(vtrip,1,1), bill_addr with vbill_add, bill_city with vbill_ci, bill_state with vbill_sta

  replace bill_zip with vbill_zip, care with vcare, caid with vcaid, pat_nbr with vpat_nbr, ;
  pat_member with vpat_member, authorizat with vauthori, type_svc with substr(vtype_svc,1,1)

  replace send_addr with accounts->send_addr, send_city with accounts->send_city, ;
  send_state with accounts->send_state, send_zip with accounts->send_zip

  replace posted with 'N', hurry with substr(vhurry,1,1), coment1 with vcoment1, coment2 with vcoment2, ;
  created with date(), hora_crea with time()

  if vsw_regu = !
    * ticket es de paciente regular
    replace nro_regupa with regu_ti->patient_nu
  endif
endif
unlock all
@ 0,0 say 'New Ticket Trip: ' + str(vticket)
return

procedure listait

```

1/20 1990

```

on key label f4 dwv = ''
define popup regular from 12,22 to 21,58 shadow title ' Search regular patient by: ' color scheme 17
define bar 1 of regular prompt ' A. Patient account nbr.' message 'To Get regular patient by account number in computer'
define bar 2 of regular prompt ' B. Patient phone number' message 'To get info by phone number'
define bar 3 of regular prompt ' C. Member Number (CAC patients)' messa 'For CAC patients that have their member number'
define bar 4 of regular prompt ' D. Medicaid nbr.'          'For medicaid patients'
define bar 5 of regular prompt ' E. Social Security'      message 'By Social Security number'
define bar 6 of regular prompt ' F. Last Name, 1st name' message 'Get patient by his (her) last name'
define bar 7 of regular prompt ' G. By patient number' message 'Search the ticket by patient number'
#define bar 7 of regular prompt '                          ' skip
#DEFINE BAR 8 OF regular PROMPT ' G. Exit' MESSAGE 'To finish y return to DOS'

```

```
on selection popup regular do consi_da
```

```
activate popup regular
```

```
on key label f4 do listait
```

```

if vsw_regu = 1
  @ 6,42 say alltrim(str(regu_ti->patient_nu)) + ' '
endif
return

```

```
procedure consi_da
```

```
* obtine codigo del paciente
```

```
define vindo obtico from 18,3 to 22,50 shadow color scheme 19
```

```
activa vindow obtico
```

```
set messa to 'Press ESC to return'
```

```
do case
```

```
case health=by patient acct. nbr
```

```
  vacctnbr = 0
```

```
  select regu_ti
```

```
  set order to patient_nu
```

```
do while .t.
```

```
  @ 1,1 say "Enter patient's acct. nbr.: " get vacctnbr picture '999999' range 0,
```

```
  read
```

```
  if lastkey() = 27
```

```
    * no hace nada y retorna
```

```
    exit
```

```
  else
```

```
    seek vacctnbr
```

```
    if found()
```

```
      do pasa_da
```

```
      deact vindow obtico
```

```
      deact popup
```

```
      return
```

```
    else
```

```
      @ 2,1 say 'Acct. nbr. not found !'
```

```
      loop
```

```
    endif
```

```
  endif
```

```
enddo
```

```
case bar() = 2
```

```
  * search by phone number
```

```

on key label f4 dwv = ''
define popup regular from 12,22 to 21,58 shadow title ' Search regular patient by: ' color scheme
define bar 1 of regular prompt ' A. Patient account nbr.' message 'To Get regular patient by account number in computer'
define bar 2 of regular prompt ' B. Patient phone number' message 'To get info by phone number'
define bar 3 of regular prompt ' C. Member Number (CAC patients)' messa 'For CAC patients that have their member number'
define bar 4 of regular prompt ' D. Medicaid nbr.' message 'For medicaid patients'
define bar 5 of regular prompt ' E. Social Security' message 'By Social Security number'
define bar 6 of regular prompt ' F. Last Name, 1st name' message 'Get patient by his (her) last name'
define bar 7 of regular prompt ' G. By patient number' message 'Search the ticket by patient number'
*define bar 7 of regular prompt ' ' skip
*DEFINE BAR 8 OF regular PROMPT ' G. Exit' MESSAGE 'To finish y return to DOS'

```

```

on selection popup regular do consi_da

```

```

activate popup regular

```

```

on key label f4 do listait
if wsv_regu = 1
  @ 6,42 say alltrim(str(regu_ti->patient_nu)) + ' '
endif
return

```

```

procedure consi_da
* obtine codigo del paciente
define windo obtico from 18,3 to 22,50 shadow color scheme 19
activa window obtico
set messa to 'Press ESC to return'
do case
case bar() = 1
  * search by patient acct. nbr
  vacctnbr = 0
  select regu_ti
  set order to patient_nu

  do while .t.
    @ 1,1 say "Enter patient's acct. nbr.: " get vacctnbr picture '999999' range 0,
    read
    if lastkey() = 27
      * no hace nada y retorna
      exit
    else
      seek vacctnbr
      if found()
        do pasa_da
        deact wind obtico
        deact popup
        return
      else
        @ 2,1 say 'Acct. nbr. not found !'
        loop
      endif
    endif
  enddo
case bar() = 2
  * search by phone number

```



```

vacctnbr = 0
select regu_ti
set order to pat_phone

do while .t.
  @ 1,1 say "Enter patient's phone nbr.: " get vacctnbr picture '999-9999' range 0,
  read
  if lastkey() = 27
    * no hace nada y retorna
    exit
  else
    seek vacctnbr
    if found()
      do pasa_da
      deact wind obtico
      deact popup
      return
    else
      @ 2,1 say 'Phone nbr. not found !'
      loop
    endif
  endif
enddo

case bar() = 3
  * search by Member number
  vacctnbr = space(18)
  select regu_ti
  set order to pat_member

do while .t.
  @ 1,1 say "Enter Member number: " get vacctnbr picture 'XXXXXXXXXXXXXXXXXXXX'
  read
  if lastkey() = 27
    * no hace nada y retorna
    exit
  else
    seek alltrim(vacctnbr)
    if found()
      do pasa_da
      deact wind obtico
      deact popup
      return
    else
      @ 2,1 say 'Member number not found !'
      loop
    endif
  endif
enddo

case bar() = 4
  * search by Medicaid number
  vacctnbr = space(18)
  select regu_ti
  set order to caid

```



```

    * no hace nada y retorna
    exit
else
    wpati_name = upper(alltrim(wlastna) + alltrim(vfirstna))
    seek wpati_name
    if found()
        do pasa_da
        deact wind obtico
        deact popup
        return
    else
        @ 2,1 say 'Patient not found !'
        loop
    endif
endif
enddo
case bar() = 7
    * search by patient number
    wpatnumber = space(15)
    select regu_ti
    set order to pat_nbr

do while .t.
    @ 1,1 say "Enter patient number: " get wpatnumber picture 'XXXXXXXXXXXXXXXXX'
    read
    if lastkey() = 27
        * no hace nada y retorna
        exit
    else
        seek alltrim(wpatnumber)
        if found()
            do pasa_da
            deact wind obtico
            deact popup
            return
        else
            @ 2,1 say 'Patient not found !'
            loop
        endif
    endif
enddo

endcase
deact window obtico

return

procedure pasa_da
* pasa los datos del maestro de regular patients to memory variables.
deactiva vindo obtico
* pero antes tiene que mostrar las posibles alternativas de from - to
vna_pa = substr(alltrim(regu_ti->iast_name),1,16) + ', ' + substr(alltrim(regu_ti->first_name),1,16)
define popup desdeha from 9,2 to 15,50 shadow title vna_pa color scheme 3

vfrom1 = 'From: ' + substr(alltrim(regu_ti->from_addr),1,15) + ' To: ' + substr(alltrim(regu_ti->to_addr),1,15)
vfrom2 = 'From: ' + substr(alltrim(regu_ti->from1_addr),1,15) + ' To: ' + substr(alltrim(regu_ti->to1_addr),1,15)
vfrom3 = 'From: ' + substr(alltrim(regu_ti->from2_addr),1,15) + ' To: ' + substr(alltrim(regu_ti->to2_addr),1,15)

```

```
vfrom4 = 'From: ' + substr(alltrim(regu_ti->from3_addr),1,15) + ' To: ' + substr(alltrim(regu_ti->to3_addr),1,15)
```

```
vmen1 = 'Press Enter to select. ESC to cancel'
```

```
define bar 1 of desdeha prompt vfrom1 message vmen1
define bar 2 of desdeha prompt vfrom2 message vmen1
define bar 3 of desdeha prompt vfrom3 message vmen1
define bar 4 of desdeha prompt vfrom4 message vmen1
```

```
on selection popup desdeha do llenada with bar()
```

```
activate popup desdeha
```

```
return
procedure llenada
parameters vbarra
if lastkey() <> 13
  return
endif
```

```
vaccount = regu_ti->acct_nbr
vcode = regu_ti->acct_nbr
vcall_by = call_by
vph_caller = ph_caller
vhora = hor_pickup
vmin = min_pickup
vam_pm = am_pm
vcall_taker = call_taker
vlast_name = last_name
vfirst = first_name
```

```
do case
case vbarra = 1
  vfrom_add = from_addr
  vfrom_ci = from_city
  vfrom_sta = from_state
  vfrom_zip = from_zip

  vto_add = to_addr
  vto_ci = to_city
  vto_sta = to_state
  vto_zip = to_zip
case vbarra = 2
  vfrom_add = from1_addr
  vfrom_ci = from1_city
  vfrom_sta = from1_stat
  vfrom_zip = from1_zip

  vto_add = to1_addr
  vto_ci = to1_city
  vto_sta = to1_state
  vto_zip = to1_zip
case vbarra = 3
  vfrom_add = from2_addr
  vfrom_ci = from2_city
  vfrom_sta = from2_stat
  vfrom_zip = from2_zip
```

```

#wsoc_sec = soc_sec
show gets
#keyboard "{PGDN}"
select tickets
wsv_regu = 1
deact popup

return

procedure gene_act
parameters vnro_cta
# genera una cuenta en el file de pacientes regulares

wpat_phol = 9999999

if len(alltrim(wpat_phone)) > 1
# convierte el numero de telef. string en numerico
wveces = 1
vlarnro = len(alltrim(wpat_phone))
vnropho = alltrim(wpat_phone)
ann = ''
do while wveces <= vlarnro
  if type(substr(vnropho,wveces,1)) = 'N'
    ann = ann + substr(vnropho,wveces,1)
  endif
  wveces = wveces + 1
enddo
if len(ann) <= 7
  wpat_phol = val(ann)
else
  wpat_phol = val(substr(ann,1,7))
endif
endif

select paramet
if flock()

  vnro_cta = paramet->nxt_patien
  replace nxt_patien with vnro_cta + 1

  select regu_ti
  append blank
  replace patient_nu with vnro_cta, acct_nbr with waccount, account with accounts->company, ;
  call_by with wcall_by, ph_caller with wph_caller, hor_pickup with whora, ;
  min_pickup with wmin, am_pm with wam_pm, call_taker with wcall_taker, ;
  last_name with wlast_name, fir:t_name with wfirst, from_addr with wfrom_add

  replace from_city with wfrom_ci, from_state with wfrom_sta, from_zip with wfrom_zip, bill_to with wbill_to, ;
  to_addr with wto_add, to_city with wto_ci, to_state with wto_sta, to_zip with wto_zip, pat_phone with wpat_phol, ;
  trip with substr(wtrip,1,1), bill_addr with wbill_add, bill_city with wbill_ci, bill_state with wbill_sta

  replace bill_zip with wbill_zip, care with wcare, caid with wcaid, pat_nbr with wpat_nbr, ;
  pat_member with wpat_member, authorizat with wauthori, type_svc with substr(wtype_svc,1,1)

  replace send_addr with accounts->send_addr, send_city with accounts->send_city, ;
  send_state with accounts->send_state, send_zip with accounts->send_zip

```

```

replace hurry with substr(whurry,1,1), coment1 with vcoment1, coment2 with vcoment2
* miles_dr_a with vremi_a, miles_dr_b with vremi_b, soc_sec with vsoc_sec

```

```

endif
select tickets
replace tickets->nro_regupa with regu_ti->patient_nu
unlock all
return

```

```

* IMPRIME tickets
* -----

```

```

define window panta from 0,0 to 23,79 none
activa wind panta

```

```

* selecciona areas de trabajo
* -----

```

```

select 1
use tickets index tickets order ticket_nbr
select 3
use paramet
select 2
use accounts index accounts order acct_nbr

```

```

select tickets

```

```

* inicializa variables
* -----

```

```

vticket = 0
vaccount = '_____'
vissue = date()
vcall_by = '_____'
vph_caller = '_____'
whora = 0
vmin = 0
vam_pm = 'p'
vcall_taker = '_____'
vlast_name = '_____'
vfirst = '_____'

```

```

vfrom_add = '_____'
vfrom_ci = '_____'
vfrom_sta = '_____'
vfrom_zip = '_____'

```

```

vto_add = '_____'
vto_ci = '_____'
vto_sta = '_____'

```

```

vto_zip = '_____'

wpat_phone = '_____'
vtrip = '1_Way'
whurry = 'Rout.'

wbill_to = '_____'
wbill_add = '_____'
wbill_ci = '_____'
wbill_sta = 'F1'
wbill_zip = '_____'

wcare = '_____'
wcaid = '_____'
wpat_nbr = '_____'
wpat_member = '_____'
wauthori = '_____'
wtype_svc = 'Amb.'

vsend_add = space(45)
vsend_ci = space(20)
vsend_sta = 'F1'
vsend_zip = space(10)
vposted = 'N'

wcomment1 = '_____'
wcomment2 = '_____'

do titulos
@ 1,0 to 1,79 double
@ 23,0 to 23,79 double
set messa to 'ESC=Cancel PgDn=Next PgUp=Previous End=Last entered ' + venter + ' = Print'

do while .t.
@ 0,0 say 'Print Ticket Trip: ' get vticket picture '9999999'
read
if lastkey() = 27
deacti windo panta
return
endif

if lastkey() = 18
if .not. bof()
skip - 1
do show_ticket
endif
loop
endif

if lastkey() = 3
if .not. eof()
skip
do show_ticket
endif
loop
endif

if lastkey() = 6

```

```

    go bottom
    do show_ticket
    loop
endif

seek wticket
if found()
    do show_ticket
    do toma_val
    exit
endif
enddo

set messa to 'Press ESC to return without printing'
@ 0,0 say 'Print Ticket Trip: ' + str(wticket)

do while .t.

    * muestra en pantalla
    * -----

    set messa to 'Press ESC to cancel'
    * menu para preguntar al finalizar ticket
    * -----
    define windo vmensa from 16,10 to 20,72 shadow color scheme 20
    wrpta = '6'
    define menu opci_fi
    define pad grabaim of opci_fi prompt "Print" at 2,11
    define pad graba of opci_fi prompt "Graphic Print." at 2,24
    define pad cancela of opci_fi prompt "Cancel" at 2,46
    *DEFINE PAD REVISA OF OPCI_FI PROMPT "Check Again" AT 2,48

    on selection pad grabaim of opci_fi do final with wrpta
    on selection pad graba of opci_fi do final with wrpta
    on selection pad cancela of opci_fi do final with wrpta
    * ON SELECTION PAD REVISA OF OPCI_FI DO final with wrpta

    * Pregunta para confirmar si graba ticket
    * -----
    define window wpre from 15,8 to 20,70 shadow color scheme 20

    activa windo wpre
    @ 0,1 say 'Ticket ready:'
    wrpta = 'R'

    do while .t.
        set messa to 'Use arrows to highlight <- , -> . ' + wenter + '! to select'
        activa menu opci_fi

        if lastkey() = 27
            deact wind wpre
            set messa to
            activa windo vmensa
            wrpta = 'N'
            @ 1,0 say 'Do you want to return without printing ticket ? (Y/N) -> ' get wrpta picture '!'
            read

```



```

deactiva windo wmensa
if wrpta = 'Y'

    deactiva windo panta

    return
else
    acti windo wpre
    @ 0,1 say 'Ticket ready:'
    loop
endif

endif

exit
enddo

deactiva windo wpre
set messa to
do case
case wrpta = 'G'
    do tick_pr with ticket_nbr
case wrpta = 'S'
    do tick_pr2 with ticket_nbr
case wrpta = 'C'
    deactiva windo panta
    return to master
otherwise
    loop
endcase
activa windo wpre
set messa to
@ 1,15 say '**** Ticket Printed ****'
wait '          Press any key to Return'
deacti windo wpre
deacti windo panta
return

```

```

enddo

```

```

return

```

```

procedure final
parameters wrpta
* para saber que opcion se ha elegido
do case
case pad() = 'GRABAIM'
    wrpta = 'G'
case pad() = 'GRABA'
    wrpta = 'S'
case pad() = 'CANCELA'
    wrpta = 'C'

```

```
case pad() = 'REVISA'
  vrpta = 'R'
```

```
endcase
deacti menu
return
```

```
procedure titulos
* muestra en pantalla
* -----
```

```
if iscolor()
  set colo to r+/b
endif
```

```
@ 0,71 say wissue
@ 2,0 say 'Acct. #: ' + waccount
```

```
@ 2,35 say 'Bill: ' + wbill_to picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ 3,35 say 'Addr: ' + wbill_add picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ 4,35 say 'City: ' + wbill_ci picture 'XXXXXXXXXXXXXXXXXXXX'
@ 4,60 say 'St: ' + wbill_sta picture 'XX'
@ 4,69 say 'Zip: ' + wbill_zip picture 'XXXXXX'
```

```
@ 2,0 say 'Call by: ' + wcall_by picture 'XXXXXXXXXXXXXXXXXXXXXX'
@ 4,0 say 'Phone : ' + wph_caller picture 'XXXXXXXXXXXXXXXXXX'
```

```
@ 5,0 to !5,79
@ 6, 32 say ' Patient: '
```

```
@ 7,1 say 'Last Name: ' + wlast_name picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ 8,1 say 'First Name: ' + wfirst picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ 9,1 say 'Pat. phone: ' + wpat_phone picture 'XXXXXXXXXXXXXXXXXX'
```

```
@ 7,54 say 'Caid: ' + wcaid picture 'XXXXXXXXXXXXXXXXXXXX'
@ 8,50 say 'Member #: ' + wpat_member picture 'XXXXXXXXXXXXXXXXXXXX'
@ 9,50 say 'Pat.Nbr.: ' + wpat_nbr picture 'XXXXXXXXXXXXXXXXXXXX'
@ 10,50 say 'Auth. #: ' + wauthori picture 'XXXXXXXXXXXXXXXXXXXX'
@ 11,50 say 'Medicare: ' + wcare picture 'XXXXXXXXXXXXXXXXXXXX'
```

```
@ 11,1 say 'From: ' + wfrom_add picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ 12,1 say 'City: ' + wfrom_ci picture 'XXXXXXXXXXXXXXXXXXXXXX'
@ 12,28 say 'State: ' + wfrom_sta picture 'XX'
@ 12,41 say 'Zip: ' + wfrom_zip picture 'XXXXXXXXXX'
```

```
@ 14,1 say 'To : ' + wto_add picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ 15,1 say 'City: ' + wto_ci picture 'XXXXXXXXXXXXXXXXXXXXXX'
@ 15,28 say 'State: ' + wto_sta picture 'XX'
@ 15,41 say 'Zip: ' + wto_zip picture 'XXXXXXXXXX'
```

```
@ 14,62 say 'Time P/U Req.'
@ 15,62 say whora picture '99'
@ 15,65 say ': '
@ 15,67 say wain picture '99'
@ 15,73 say 'Am/Pm'
@ 15,71 say wam_pm
```



```

@ 15,7 say to_city picture 'XXXXXXXXXXXXXXXXXXXXX'
@ 15,35 say to_state picture 'XX'
@ 15,46 say to_zip picture 'XXXXXXXXXX'

```

```

+ @ 14,62 say 'Time P/U Req.'
@ 15,62 say hor_pickup picture '99'
@ 15,65 say ':'
@ 15,67 say min_pickup picture '99'
+ @ 15,73 say 'Am/Pm'
@ 15,71 say alltrim(am_pm) + 'm'

```

```

do case
case trip = '1'
  vtrip = '1_Way'
otherwise
  vtrip = 'RTrip'
endcase

```

```

do case
case hurry = 'R'
  whurry = 'Rout.'
otherwise
  whurry = 'ASAP'
endcase

```

```

do case
case type_svc = 'A'
  vtype_svc = 'Amb.'
case type_svc = 'H'
  vtype_svc = 'W/C'
otherwise
  vtype_svc = 'Str.'
endcase

```

```

@ 17,12 say call_taker picture 'XXXXXXXXXXXXXXXXXXXXX'

```

```

@ 19,18 say vtrip
@ 19,49 say vtype_svc
@ 19,73 say whurry

```

```

@ 21,8 say coment1 picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ 22,7 say coment2 picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'

```

```

if iscolor()
  set colo to gr+/b
endif
return

```

```

procedure toma_val
+ =====+

```

```

vticket = ticket_nbr
vaccount = tickets->acct_nbr
vissue = date_issue
vcall_by = call_by
vph_caller = ph_caller
vhora = hor_pickup
vmin = min_pickup
vam_pm = am_pm
vcall_taker = call_taker
vlast_name = last_name
vfirst = first_name

```

```

wfrom_add = from_addr
wfrom_ci = from_city
wfrom_sta = from_state
wfrom_zip = from_zip

```

```

vto_add = to_addr
vto_ci = to_city
vto_sta = to_state
vto_zip = to_zip

```

```

wpat_phone = pat_phone
do case
case trip = '!'
  wtrip = '!_Way'
otherwise
  wtrip = 'Rtrip'
endcase

```

```

do case
case hurry = 'R'
  whurry = 'Rout.'
otherwise
  whurry = 'ASAP'
endcase

```

```

wbill_to = bill_to + space(20)
wbill_add = bill_addr + space(30)
wbill_ci = bill_city + space(30)
wbill_sta = bill_state
wbill_zip = bill_zip

```

```

wcare = care
wcaid = caid
wpat_nbr = pat_nbr
wpat_member = pat_member
wauthori = authorizat

```

```

do case
case type_svc = 'A'
  wtype_svc = 'Amb.'
case type_svc = 'W'
  wtype_svc = 'W/C'
otherwise
  wtype_svc = 'Str.'
endcase

```

```

*wsend_add = space(45)
*wsend_ci = space(20)
*wsend_sta = 'F1'
*wsend_zip = space(10)
wposted = 'N'

```

```

wcoment1 = coment1
wcoment2 = coment2

```

```

return

```

```

* prg: chk_invo
* goal: to check invoices, add mileage to the invoices.
* -----

```

```

define window panta from 0,0 to 23,79 none
activa wind panta

```

```

define windo umensa from 15,5 to 20,75 shadow color scheme 19

```

```

define window invo_win from 12,0 to 20,79 color scheme 23

```

```

* menu para preguntar al finalizar invoice
* -----

```

```

define window wpre from 18,5 to 20,75 shadow color scheme 19

```

```

wrpta = '6'

```

```

define menu opci_fi color scheme 18

```

```

define pad salva of opci_fi prompt "Save" at 0,10

```

```

* DEFINE PAD SALVAIN OF OPCFI PROMPT "Save and Print" AT 0,12

```

```

define pad iteas of opci_fi prompt "Add items" at 0,24

```

```

define pad revisa of opci_fi prompt "Other Invoice" at 0,40

```

```

on selection pad salva of opci_fi do final with wrpta

```

```

* ON SELECTION PAD SALVAIN OF OPCFI DO final with wrpta

```

```

on selection pad iteas of opci_fi do final with wrpta

```

```

on selection pad revisa of opci_fi do final with wrpta

```

```

* selecciona areas de trabajo
* -----

```

```

select !4

```

```

use regu_ti index regu_ti order patient_nu

```

```

select !

```

```

use tickets index tickets order ticket_nbr

```

```

select 3

```

```

use paramet

```

```

select 2

```

```

use accounts index accounts order acct_nbr

```

```

select 4

```

```

use rates index rates order acc_svc

```

```

select 5

```

```

use services index services order svc_nbr
select 6
use invoices index invoices order ticket_nbr

select tickets

do titulos
vticket = 0
@ 23,0 to 23,79 double
set messa to 'ESC=Cancel   PgDn=Next   PgUp=Previous   End=Last entered   ' + venter + ' = Check'

do while .t.
  @ 0,50 say 'Invoice Nbr.: ' get vticket picture '9999999'
  read
  if lastkey() = 27
    deacti windo panta
    return
  endif

  if lastkey() = 18
    if .not. boff()
      skip - !
      do show_invo
    endif
  loop
endif

  if lastkey() = 3
    if .not. eof()
      skip
      if .not. eof()
        do show_invo
      endif
    endif
  loop
endif

  if lastkey() = 6
    seek paramet->last_invo
    if found()
      do show_invo
    endif
  loop
endif

  seek vticket
  if found()
    do show_invo
    do miles
  endif
  set messa to 'ESC=Cancel   PgDn=Next   PgUp=Previous   End=Last entered   ' + venter + ' = Check'
enddo

return

```

```

*****
* FIN DE PROGRAMA PRINCIPAL
* -----

procedure miles

if date_issue >= date()
  * advierte si es que la fecha del ticket es todays' date
  * -----
  activa windo wmensa
  set messa to 'Press ESC to return'
  @ 0,0 say "You should not process invoices with today's date.!"
  wrpta = 'N'
  @ 2,0 say 'Do you really want to continue ? (Y/N):' get wrpta picture '!'
  read
  deactiva windo wmensa
  if lastkey() = 27 .or. wrpta (<) 'Y'
    return
  endif
endif

do while .t.
  set messa to 'ESC to finish. "*" in CODE field to delete items.'
  @ 0,50 say 'Invoice Nbr.: ' + str(wticket,7,0)
  serviced = date_issue
  @ !0,33 get serviced

  if len(a!trim(tickets->caid)) (<) 0
    * Si es medicaid se necesita: activo, hora de inicio, fin servicio
    * origin and destination
    * -----
    @ 10,54 get tickets->autho_by picture 'XXXXXXXXXXXXXXXXXXXXX'
    @ !!,7 get tickets->start
    @ !!,13 get tickets->start_ho picture '@M A,P'
    @ !!,24 get tickets->stop
    @ 11,30 get tickets->stop_ho picture '@M A,P'
    @ 11,44 get tickets->active picture 'XXXXXXXXXXXXXXXXXXXXX'
    @ 12,8 get tickets->caid_orig
    @ 12,38 get tickets->caid_dest
  endif
  read
  if tickets->nro_regupa > 1
    select regu_ti
    seek tickets->nro_regupa
    @ !!,45

    if found()
      @ 11,58 say 'Miles:'
      wiles = regu_ti->miles_d?_a + regu_ti->miles_d?_b
      wiles! = regu_ti->miles!_a + regu_ti->miles!_b
      wiles2 = regu_ti->miles2_a + regu_ti->miles2_b
      wiles3 = regu_ti->miles3_a + regu_ti->miles3_b

      @ 11,66 say wiles picture '99' color +w/b
      @ !!,68 say ', '
      @ !!,70 say wiles! picture '99' color +w/b
      @ !!,73 say ', '
    endif
  endif
endif

```



```

    @ 11,74 say wmailes2 picture '99'    color ++w/b
    @ 11,76 say ', '
    @ 11,77 say wmailes3 picture '99'    color ++w/b

endif
else
  @ 11,45
endif
*****MAG.i
warchi = 'ga' + str((1 + rand(-2)) * 100000,6,0)
wfiledbf = warchi + '.dbf'
select invoices
copy structure to &wfiledbf with cdx
select 15
use &warchi index &warchi order ticket_nbr

select invoices
set order to ticket_nbr
seek wticket
**
do while invoices->ticket_nbr = wticket .and. ( .not. eof() )
  select &warchi
  append blank
  replace ticket_nbr with invoices->ticket_nbr
  replace svc_nbr with invoices->svc_nbr, des_svc with invoices->des_svc
  replace qty with invoices->qty, rate with invoices->rate
  select invoices
  skip
enddo
*****

select &warchi

browse fields qty :v= qty >= 0 :p = '9999.99' :h = ' Qty.' , ;
  svc_nbr :v= lista(&warchi->svc_nbr) :h= ' Code ' , ;
  des_svc :h = ' Description', ;
  rate :v= rate >= 0 :p = '9999.99' :h = ' Rate ' , ;
  wparcial = qty * rate :r :p = '99999.99' ;
  for ticket_nbr = tickets->ticket_nbr ;
  nomenu noappend nodelete title ' Items ' window invo_win

set filter to
*****

select invoices
set order to ticket_nbr
seek wticket
**
if found()
  do while ticket_nbr = wticket
    delete
    skip
  enddo
endif
*****

select &warchi

```

```

set order to ticket_nbr
seek vticket
**
if found()
**
do while ticket_nbr = vticket .and. (.not. eof() )
select invoices
append blank
replace ticket_nbr with &warchi->ticket_nbr
replace svc_nbr with &warchi->svc_nbr, des_svc with &warchi->des_svc
replace qty with &warchi->qty, rate with &warchi->rate
select &warchi
skip
enddo
endif
****
use
erase warchi + '.dbf'
erase warchi + '.cdx'
*****MAG.f

*
* comentario al invoice
* -----
set messa to 'Press ESC to finish. ' + wenter + 'next line.'
activa windo wmensa
select tickets
@ 0,0 say 'Any comment on this invoice ?'
@ 1,1 get tickets->comment1
@ 2,1 get tickets->comment2
read
deactiva windo wmensa

do show_invo

set messa to 'Press ESC to cancel!'

* Pregunta para confirmar si graba ticket
* -----

activa windo wpre
vrpta = 'R'

do while .t.
set messa to 'Use arrows to highlight (- , -) . ' + wenter + ' to select'
activa menu ooci_fi

if lastkey() = 27
deact wind wpre
set messa to
activa windo wmensa
vrpta = 'N'
@ 1,0 say 'Do you want to return without saving ticket ? (Y/N) -> ' get vrpta picture !!'
read

```

```

deactiva windo waensa
if wrpta = 'Y'

    deactiva windo panta

    return to master
else
    acti wind wpre
    loop
endif

endif

exit
enddo

deactiva windo wpre
set messa to
do case
case wrpta = 'G'
    * salva y continua al sgte.
    * -----
    replace tickets->posted with 'Y', tickets->pynt with 0, tickets->tot_invo with wtot_in , tickets->billed with date()

case wrpta = 'S'
    * salva e imprime
    * -----
    replace tickets->posted with 'Y', tickets->pynt with 0, tickets->tot_invo with wtot_in , tickets->billed with date()

case wrpta = 'C'
    * modifica items
    * -----
    do otro_item
    select tickets
    loop
case wrpta = 'O'
    * retorna al inicio de ingresar ticket
    * -----
otherwise
    loop
endcase
activa windo wpre
set messa to
deacti windo wpre
@ !0,8 say billed
select paramet
replace last_invo with tickets->ticket_nbr
select tickets
exit

enddo
return

procedure final
parameters wrpta
* para saber que opcion se ha elegido

```

```

do case
case pad() = 'SALVA'
  vrpta = 'G'
case pad() = 'SALVAIN'
  vrpta = 'S'
case pad() = 'ITEMS'
  vrpta = 'C'
case pad() = 'REVISA'
  vrpta = 'D'

endcase
deacti menu
return

procedure modi_ti
* =====
set messa to 'Please wait..... saving ticket'
select accounts
set order to acct_nbr
seek vaccount

select tickets
if posted = 'Y'
  activa windo vpre
  set messa to
  @ 1,19 say 'Ticket already billed. modify denied.'
  wait '          Press any key to Return'
  deacti windo vpre
  deactiva windo panta
  return to master
endif

if flock()

  replace ticket_nbr with vticket, acct_nbr with vaccount, tickets->account with accounts->company, date_issue with wissue,
  call_by with wcall_by, ph_caller with wph_caller, hor_pickup with whora, ;
  sin_pickup with vmin, am_pm with wam_pm, call_taker with wcall_taker, bill_to with wbill_to, ;
  last_name with wlast_name, first_name with wfirst, from_addr with wfrom_addr

  replace from_city with wfrom_ci, from_state with wfrom_sta, from_zip with wfrom_zip, ;
  to_addr with wto_add, to_city with wto_ci, to_state with wto_sta, to_zip with wto_zip, pat_phone with wpat_phone, ;
  trip with substr(wtrip,1,!), bill_addr with wbill_add, bill_city with wbill_ci, bill_state with wbill_sta

  replace bill_zip with wbill_zip, care with wcare, caid with wcaid, pat_nbr with wpat_nbr, ;
  pat_member with wpat_member, authorizat with wauthori, type_svc with substr(wtype_svc,1,!)

  replace send_addr with accounts->send_addr, send_city with accounts->send_city, ;
  send_state with accounts->send_state, send_zip with accounts->send_zip

  replace posted with 'N', hurry with substr(whurry,!,!), coment1 with wcoment1, coment2 with wcoment2

endif
unlock all
*@ 0,0 say 'Ticket Trip: ' + str(vticket)
return

```

```

procedure titulos
* muestra en pantalla
* -----
if iscolor()
  set colo to r+/b
endif

@ 0,0 say em_name

@ 1,0 to 9,79

@ 5,1 to 5,78

@ 2,40 to 4,40
@ 6,45 to 8,45
@ 1,7 say ' To: '

@ 1,55 say ' Ship To: '

@ 5,7 say ' Patient: '

@ 5,55 say ' Others: '

@ 13,0 to 13,79 double
@ 15,0 to 15,79

@ 14,0 say 'Qty.'
@ 14,19 say 'Description'
@ 14,55 say 'Date'
@ 14,70 say 'Amount'

@ 21,0 to 21,79 double
@ 22,60 say 'TOTAL: '

if iscolor()
  set colo to gr+/b
endif
return

procedure show_invo
* =====*
wticket = ticket_nbr

if iscolor()
  set colo to w+/b
endif

if tickets->checked <> 'Y'
  do crea_invo

```

```

endif
@ 2,1 clear to 4,39
select accounts
set order to acct_nbr
seek alltrim(tickets->acct_nbr)

if found()
  @ 2,1 say alltrim(tickets->acct_nbr) + ' - '
  @ 2,9 say tickets->account picture 'XXXXXXXXXXXXXXXXXXXXX'
  @ 3,1 say alltrim(accounts->address) picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
  @ 4,1 say alltrim(accounts->city) + ' ' + accounts->state + ' ' + accounts->zip picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
else
  @ 2,1 say tickets->acct_nbr
  @ 3,1 say 'Account not found !!'
endif

select tickets
@ 2,4! say bill_to picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ 3,4! say alltrim(bill_addr) picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ 4,4! say alltrim(bill_city) + ' ' + bill_state + ' ' + bill_zip picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'

* patient
* -----
@ 6,1 clear to 8,44
@ 6,1 say alltrim(last_name) + ' ' + alltrim(first_name) picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
@ 7,1 say 'Fr: ' + alltrim(from_addr) + ' To: ' + alltrim(to_addr) picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
if trip = 'R'
  @ 8,1 say 'Fr: ' + alltrim(to_addr) + ' To: ' + alltrim(from_addr) picture 'XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX'
endif

* others
* -----

wfile = 5
@ wfile,46 clear to 8,78
if len(alltrim(care)) <> 0
  @ wfile,46 say 'Care: ' + alltrim(care)
  wfile = wfile + 1
endif

if len(alltrim(caid)) <> 0
  @ wfile,46 say 'Caid: ' + alltrim(caid)
  wfile = wfile + 1
endif

if len(alltrim(pat_nbr)) <> 0
  @ wfile,46 say 'Pat.#: ' + alltrim(pat_nbr)
  wfile = wfile + 1
endif

if len(alltrim(pat_member)) <> 0
  @ wfile,46 say 'Pat.Member: ' + alltrim(pat_member)
  wfile = wfile + 1
endif

if len(alltrim(authorizat)) <> 0

```

```

    @ wfile,46 say 'Authoriz.: ' + alltrim(authorizat)
    wfile = wfile + 1
endif

* fechas y otros
* -----
@ 10,0
@ 11,0
@ 12,0

@ 10,0 say 'Billed:'
@ 10,8 say billed

@ 10,22 say 'Serviced:'
@ 10,23 say serviced

* si es medicaid muestra otros datos
* -----

if len(alltrim(tickets->caid)) <> 0
    @ 10,45 say 'Auth.By: ' + tickets->autho_by
    @ 11,0 say 'Start: ' + tickets->start
    @ 11,13 say tickets->start_ho

    @ 11,18 say 'Stop: ' + tickets->stop
    @ 11,30 say tickets->stop_ho

    @ 11,36 say 'Reason: ' + tickets->motive
    @ 12,0 say 'Origin: ' + tickets->caid_orig
    @ 12,31 say 'Dest.: ' + tickets->caid_dest
endif

* muestra items del invoice
* -----

@ 16,0 clear to 20,79
select invoices
seek tickets->ticket_nbr
if found()
    wfile = 16
    wsubtot = 0
    do while (invoices->ticket_nbr = tickets->ticket_nbr) .and. (.not. eof()) .and. ;
        wfile <= 20

        @ wfile, 1 say invoices->qty
        @ wfile, 8 say alltrim(invoices->svc_nbr) + ' - ' + invoices->des_svc
        @ wfile, 50 say invoices->rate picture '9999.99'
        wcant = round(invoices->qty * invoices->rate,2)
        wsubtot = wsubtot + wcant
        @ wfile, 70 say wcant picture '99999.99'
        wfile = wfile + 1
    skip
enddo
@ 22,3 clear to 22,55

```

```
if wfila > 20
```

```
  * en caso de haber mas items que no se pueden presentar en pantalla
```

```
  * -----
```

```
  do while (invoices->ticket_nbr = tickets->ticket_nbr) .and. ( .not. eof() )
```

```
    vcant = round(invoices->qty * invoices->rate,2)
```

```
    vsubtot = vsubtot + vcant
```

```
    wfila = wfila + 1
```

```
    skip
```

```
  enddo
```

```
  @ 22,3 say '(Partial list of items)'
```

```
endif
```

```
@ 22,70 say vsubtot picture '99999.99'
```

```
vtot_in = vsubtot
```

```
else
```

```
@ wfila + 2, 30 say 'No se encontro items !!!'
```

```
endif
```

```
select tickets
```

```
if iscolor()
```

```
  set colo to gr+/b
```

```
endif
```

```
return
```

```
procedure crea_invo
```

```
* =====*
```

```
* -----
```

```
* aumenta registros en invoices.dbf segun el tipo de servicio (type_svc)
```

```
* en tickets.dbf
```

```
* wserv = es el servicio basico (response)
```

```
* wserv2 = es el costo por milla ( despues de la milla 5)
```

```
* -----
```

```
select tickets
```

```
wserv = '00000'
```

```
wserv2 = '00001'
```

```
do case
```

```
case tickets->type_svc = 'A'
```

```
  wserv = 'AM001'
```

```
  wserv2 = 'AM002'
```

```
case tickets->type_svc = 'W'
```

```
  wserv = 'AM004'
```

```
  wserv2 = 'AM005'
```

```
case tickets->type_svc = 'S'
```

```
  wserv = 'AM007'
```

```
  wserv2 = 'AM008'
```

```
endcase
```

```
wdescri = 'Service'
```

```
wdescri2 = wdescri + ' rate per mile'
```

```
select services
```



```

seek wserv
if found()
  wdescri = alltrim(services->svc_descr)
endif
seek wserv2
if found()
  wdescri2 = alltrim(services->svc_descr)
endif

wcodigo = alltrim(tickets->acct_nbr) + alltrim(wserv)
select rates
seek wcodigo
if found()
  select invoices

  * crea response charge
  * -----
  append blank
  replace invoices->ticket_nbr with tickets->ticket_nbr, invoices->svc_nbr with wserv, ;
    invoices->des_svc with wdescri, invoices->rate with rates->svc_base

  if tickets->trip = 'R'
    replace invoices->qty with 2
  else
    replace invoices->qty with 1
  endif
else
  activa wind invo_win
  @ 0,0 say 'I could not find rate for next invoice'
  @ 1,0 say 'Creating generic invoice, check rates !'
  wait
  deacti wind invo_win
  select invoices
  append blank
  replace invoices->ticket_nbr with tickets->ticket_nbr, invoices->svc_nbr with wserv, ;
    invoices->des_svc with wdescri, invoices->rate with 0
  replace invoices->qty with 1
endif

wcodigo = alltrim(tickets->acct_nbr) + alltrim(wserv2)
select rates
seek wcodigo
if found()
  * crea registro por mileage
  * -----
  select invoices
  append blank
  replace invoices->ticket_nbr with tickets->ticket_nbr, invoices->svc_nbr with wserv2, ;
    invoices->des_svc with wdescri2, invoices->rate with rates->svc_base
  replace invoices->qty with 1
else

  activa wind invo_win
  @ 0,0 say 'I could not find rate for next invoice'
  @ 1,0 say 'Creating generic invoice, check rates !'
  wait

```

```

deacti wind invo_win
select invoices
append blank
replace invoices->ticket_nbr with tickets->ticket_nbr, invoices->svc_nbr with userv2, ;
    invoices->des_svc with wdescri2, invoices->rate with 0
replace invoices->qty with 1
endif

```

```

select tickets
whora = str(tickets->hor_pickup,2,0) + ':' + str(min_pickup,2,0)
replace tickets->checked with 'Y', tickets->motive with 'DISCHARGE', ;
    tickets->start with whora, tickets->start_ho  with am_pm, tickets->serviced with date_issue

```

```

return

```

```

function lista
parameters wcode
if substr(alltrim(wcode),1,1) = '*'
    delete
    go top
    return
endif
select services
seek alltrim(wcode)
if .not. found()
    set order to svc_descr
    define popup taller! from 3,32 to 20,70 prompt field svc_descr ;
        message 'Arrows to move, or first letter., ' + wenter + ' to select' shadow color scheme 17
    on selection popup taller! do ruteata with prompt()
    activate popup taller!
    set order to svc_nbr

```

```

endif

```

```

select rates
wnumerin = alltrim(tickets->acct_nbr) + alltrim(services->svc_nbr)
seek wnumerin
if found()
    replace &warchi->rate with rates->svc_base
else
    replace &warchi->rate with 0
endif
replace &warchi->svc_nbr with services->svc_nbr
replace &warchi->des_svc with services->svc_descr.

```

```

show get &warchi->svc_nbr
show get &warchi->des_svc
show get &warchi->rate

```

```

select &warchi
return .t.

```

```
procedure ruteata
parameters empresa
valor = 0
```

```
seek empresa
if .not. found()
    return
else
    valor = !
    if lastkey() = 13
        deacti popup
        waccount = services->svc_nbr
    endif
endif
return
```

```
procedure otro_item
* agrega un item al invoice
* -----
select invoices
append blank
replace invoices->ticket_nbr with tickets->ticket_nbr, invoices->svc_nbr with '_____',
    invoices->des_svc with '_____', invoices->rate with 0
replace invoices->qty with 1
return
```

EASTERN BUSINESS FORMS, INC. (305) 871-8193

C-25773-F

TICKET NUMBER:

Pick/Up Date

P/U Time

CALL RECEIVED

-Patient

01

-Patient Phone

03

-From

04

-To

03

04

-Transportation

05

BILL TO:

Response Charge

Waiting Time

Care:
Caid:

Special Handling

Pat. #:
PtMem:

Miles

X =

TOTAL

Auth #:

UNIT NUMBER





INVOICE

INVOICE NO:
INVOICE DATE:



3204 N.W. 79 Ave.
Miami, Fl. 33122

Tel.: (305) 599-9433
Fax: (305) 599-9413

ACCOUNT NO:
BILL TO:

DATE OF SERVICE:
PATIENT NAME:
NUMBER:
AUTHORIZATION NO:

EASTERN BUSINESS FORMS, INC. (305) 871-8193

TRANSPORTATION			
QTY.	SERVICE	RATE	AMOUNT
THANK YOU			TOTAL

TC-25772-F



1111



131 - A

1. A = ADJUSTMENT V = VOID

2. TRANSACTION CONTROL NUMBER



REQUEST FOR PAYMENT

FLORIDA MEDICAID PROGRAM
DEPARTMENT OF HEALTH AND REHABILITATIVE SERVICES
TRANSPORTATION SERVICES

RETURN TO:

CONSULTEC, INC.
P.O. BOX 7052
TALLAHASSEE, FLORIDA 32314-7052

3. PROVIDER NAME & ADDRESS

4. PRIOR AUTHORIZATION NO 0807	5. PROVIDER NO. 634917	7. BILLING DATE
-----------------------------------	---------------------------	-----------------

8. RECIPIENT'S NAME (LAST, FIRST, MI)	9. RECIPIENT'S MEDICAID ID #	10. DATE OF BIRTH	11. SEX	12. PATIENT STATUS
---------------------------------------	------------------------------	-------------------	---------	--------------------

13.	14.	15.	16. TRIP INDICATOR 2. = NON-EMERGENCY (SEE 16B) 3. = NON-EMERGENCY (SEE 16B)
-----	-----	-----	--

17. NAME OF PERSON IN DISTRICT OFFICE AUTHORIZING TRANSPORTATION	18. DESCRIBE REASON TRANSPORTATION SERVICES PROVIDED	19. NAME OF PERSON REQUESTING TRANSPORTATION
--	--	--

OTHER PAYMENT SOURCE	20. OTHER MEDICAL INSURANCE CODE	21. INSURANCE POLICY NUMBER
	22. INSURANCE COMPANY'S NAME	23. INSURANCE COMPANY'S ADDRESS (CITY & STATE)

STATEMENT OF SERVICES

24.	25. ORIGINATING LOCATION/FACILITY	26. ORIGINATING LOCATION ADDRESS
-----	-----------------------------------	----------------------------------

27.	28. DESTINATION LOCATION/FACILITY	29. DESTINATION LOCATION ADDRESS
-----	-----------------------------------	----------------------------------

30. DATE OF SERVICE	31. TIME OF SERVICE START <input type="checkbox"/> A.M. <input type="checkbox"/> P.M. STOP <input type="checkbox"/> A.M. <input type="checkbox"/> P.M.	32. NUMBER OF ADDITIONAL MEDICAID PATIENTS SHARING RIDE
---------------------	---	---

33.	34. PROCEDURE CODE	35. PROCEDURE CODE DESCRIPTION	36.	38. USUAL AND CUSTOMARY CHARGE	37. PAID BY PRIMARY CARRIER
1	A0150	BASE RATE - AMBULANCE			
2	A0130	BASE RATE - WHEELCHAIR VAN/DRIVER			
3	W9885	BASE RATE - WHEELCHAIR VAN/DRIVER/ATTENDANT			
4	W9886	MILEAGE CHARGE - VAN	NO OF MILES	RATE/MI.	
5	A0020	MILEAGE CHARGE - AMBULANCE	NO OF MILES	RATE/MI.	
6	A0070	OXYGEN	OXY MIN.	OXY RATE	
7					
8					
9					
10					

IMPORTANT

READ CERTIFICATION STATEMENT ON REVERSE SIDE.

THE PERSON WHOSE SIGNATURE APPEARS BELOW CERTIFIES THAT HE/SHE IS AUTHORIZED TO SIGN THIS INVOICE AND THAT HE/SHE HAS READ AND WILL COMPLY WITH ALL OF THE TERMS AND CONDITIONS WHICH ARE CONTAINED IN THE CERTIFICATION STATEMENT WHICH APPEARS ON THE REVERSE SIDE OF THIS INVOICE.

AUTHORIZED SIGNATURE/TITLE
NON-EMER VAN (1-89)

DATE

RETURN WHITE COPY TO CONSULTEC, INC

FOR OFFICE USE ONLY

38. TOTAL CHARGED

F9: FONDE

9.5 Beneficio - Costo

La relación beneficio-costo es un indicador de rentabilidad, que relaciona el flujo neto de fondos actualizado, durante el horizonte del proyecto con la inversión inicial. Esta relación se puede expresar en la siguiente fórmula:

$$\text{Relac. Benef.-Costo (RB/C)} = \frac{\text{Valor Actual Neto}}{\text{Inversión Inicial}}$$

Si RB/C es mayor o igual a 1 el proyecto es aceptado
 Si RB/C es menor que 1 el proyecto es rechazado, el proyecto en realidad no es rentable.

En el caso de la instalación de la red de computadores de la red Novell 3.11 y el desarrollo de la aplicación principal se había estimado un horizonte de 6 años. La inversión inicial se había calculado en U.S. \$ 57,000. En el cuadro de totales de la página 10 adicionar U.S. \$ 20,000 por concepto de contratación de la empresa consultora para la elaboración de la aplicación en la red de computadores, personal adicional y gastos administrativos. Depreciación de las computadores en 3 años.

El valor actual neto del flujo anual de fondos se había calculado en U.S. \$ 185,000. Considerandose en este estimado una entrada adicional de 30 nuevas facturas diarias debido a este proyecto por un valor de U.S. \$ 20.00 a 30.00 por factura. Por lo tanto el ratio de Beneficio - Costo es de:

$$\text{RB/C} = \frac{185,000}{57,000} = 3.24 > 1 \quad (\text{el proyecto se acepta})$$

Desde que se implemento el sistema hace mas de 2 años, el retorno de la inversion inicial ha excedido los cálculos estimados, es decir la estimación había sido conservadora ya que por ejemplo en el 2do año de operación no se esta facturando 30 nuevas facturas al día sino que se ha duplicado, alrededor de 65 facturas adicionales al día, lo que hace la inversión sumamente rentable.

9.6 Bibliografia

- Mastering Foxpro, Advanced Techniques, Joseph Lackner, 1992, published by QUE.

Novell's Guide to managing small Netware networks, 1990, 1991, published by Novell Press.

Novell's Problem-solving guide for Netware Systems, 1991, published by Novell Press.

- Object-Oriented analysis - Peter Coad and Edward Yourdon, 1991, Yourdon Press/ Prentice Hall

Novell's Guide to integrating UNIX and netware Networks, 1991, published by Novell Press.

- IBM, General purpose of the IBM/AS400 System, 1991

- Technical manuals about Lantastic operating Systems, 1989, published by Artisoft, Inc. Ca.

Vines the operating system for WAN (Wide Area Networks), 1990, technical manuals published by Banyan Corp.

Corporate Computing, Ziff Publishing Company, monthly publication, issues from May to October '92. 'Comparative studies about several operating systems'

Contabilidad de Costos, Un enfoque administrativo y de gerencia, 1989. Morton Backer, Lyle Jacobsen.

- Data Base Advisor, published by Data base solutions, Monthly publication about client/server in personal computers, 1991, 1993.

Netware Connection, Published by Netware Users International, January to July issues 1991, Bimonthly publication about Novell products.