

UNIVERSIDAD INTERNACIONAL DEL ECUADOR

Facultad de Ingeniería en Mecánica Automotriz

TESIS DE GRADO PARA LA OBTENCION DEL TITULO DE

INGENIERO EN MECANICA AUTOMOTRIZ

**Implementación de un Software para el control, administración y gestión de
calidad de talleres automotrices**

Andrés Fernando Novillo Espinosa

Director: Ing. José Andrés Castillo

2010

Quito, Ecuador

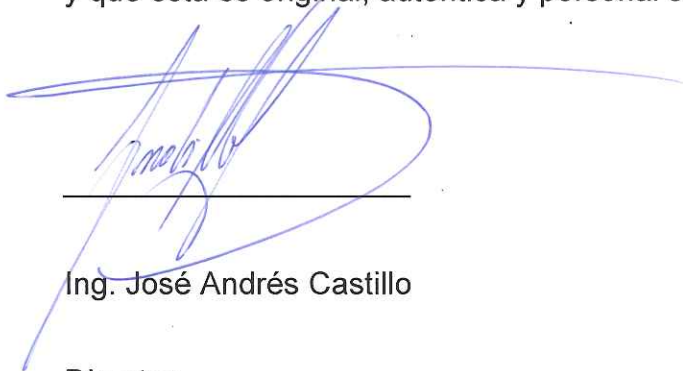
CERTIFICACIÓN

Yo, Andrés Fernando Novillo Espinosa, declaro que soy el autor exclusivo de la presente investigación y que ésta es original, auténtica y personal mía. Todos los efectos académicos y legales que se desprendan de la presente investigación serán de mi exclusiva responsabilidad.



Andrés Fernando Novillo Espinosa

Yo, José Andrés Castillo, declaro que, en lo que yo personalmente conozco, el señor, Andrés Fernando Novillo Espinosa, es el autor exclusivo de la presente investigación y que ésta es original, auténtica y personal suya.



Ing. José Andrés Castillo

Director

DEDICATORIA

Este trabajo teórico y práctico de tesis lo dedico primero a Dios, quien ha estado y está siempre conmigo, y quien ha sido mi fortaleza para luchar por alcanzar mis sueños. Además, esto lo dedico a mi hermosa familia y a mi novia, quienes con paciencia, apoyo y amor han vivido conmigo el desarrollo de mi carrera y quienes a pesar de los obstáculos, estuvieron ahí en cada paso que di y siempre tuvieron fe en mí.

ANDRÉS

AGRADECIMIENTO

Quiero manifestar mi eterna gratitud a la persona que ha sido mi guía para la culminación de esta mi carrera, mi profesor Ingeniero José Andrés Castillo, que con su dedicación, voluntad férrea y alegría por compartir sus conocimientos, hizo posible que yo pudiera terminar este proyecto de tesis.

A mis compañeros y profesores que me acompañaron a lo largo de mi carrera y quienes han marcado grandes enseñanzas en mi vida. Pero, sobretodo, agradezco a la universidad, la cual me brindo grandes oportunidades, siendo la más grande la beca deportiva, con la cual pude estudiar toda mi carrera.

Quedo, eternamente agradecido.

ANDRÉS

ÍNDICE

Resumen	XII
Abstract	XV
Introduccion	XVIII
Capítulo 1	1
1.1. Marco Teórico	1
1.1.1. Concepto De Calidad	1
1.1.2. Historia De La Calidad.....	1
1.1.3. Herramientas Estadísticas Para La Calidad	2
1.1.3.1. Hoja De Control.....	2
1.1.3.2. Histograma	2
1.1.3.3. Diagrama De Pareto.....	2
1.1.3.4. Diagrama De Causa - Efecto.....	3
1.1.3.5. Diagramas De Flujo.....	3
1.1.3.6. Diagramas De Dispersión.....	3
1.1.4. Administración Por Calidad	3
1.1.5. Mejoramiento Continuo	4
1.1.6. Gestión De La Calidad	4
1.1.7. Software Como Herramienta De Control De Calidad	5

1.1.8. Visual Studio.Net.....	5
1.1.9. ¿Qué Es Visual Basic.Net?	6
1.1.10. Entorno De Desarrollo.....	6
Capítulo 2.....	8
2.1. Desglose De Actividades Existentes	9
2.2. Planeamiento De Nuevos Organigramas Para Administración De Taller.....	10
2.3. Propuesta Perfiles Laborales	11
2.3.1. Detalle De Las Funciones De Cada Responsable	11
2.3.1.1. Jefe De Taller Mecánico.....	11
2.3.1.2. Asesor de servicio	13
2.3.1.3. Coordinador De Mantenimiento	14
2.3.1.4. Técnico Especializado.....	15
2.3.1.5. Técnico De Mantenimiento.....	15
2.3.1.6. Almacén De Herramientas	16
Capitulo 3.....	17
3.1. Tipo De Empresa A La Que Se Aplica El Programa	17
3.2. Información Para El Registro De Órdenes De Trabajo.....	17
3.3. Manejo Del Catálogo De Un Catálogo De Clientes	18
3.4. Manejo De Inventario De Repuestos	18

3.5. Manejo De Reportes	18
3.6. Seguridad De La Información Que Se Maneja	19
3.7. Identificación De Las Actividades Administrativas.....	19
3.8. Clasificación Y Organización De Actividades	21
3.8.1. Actividades Administrativas De Trabajo En El Taller.....	21
3.8.1.1. Atención A Órdenes De Trabajo.....	21
3.8.1.2. Asignación De Trabajo	22
3.8.1.3. Desarrollo Y Ejecución De La Orden De Trabajo	22
3.8.1.4. Actividad Terminada.....	22
3.8.1.5. Archiva Información Y Elabora Reportes Mensuales	23
3.8.1.6. Analiza Reportes Trimestrales Para La Toma De Decisiones.....	23
3.8.2. Descripción De Cada Una De Las Etapas De Compras De Herramientas Materiales Y Consumibles.....	23
3.8.2.1. Solicitud De Compra Ya Sea De Materiales, Herramientas O Consumibles	23
3.8.2.2. Evaluación De Las Peticiones Para Consumibles Y De Las Cotizaciones...	24
3.8.2.3. Etapa De Adquisición De Las Herramientas, Materiales Y Consumibles	24
3.8.2.4. Recepción Del Producto.....	24
3.9. Implementación Del Software.....	24
3.10. Funcionamiento Del Sistema.....	28

3.10.1. Menú De Archivo	31
3.10.1.1. Cambiar Contraseña	31
3.10.1.2. Cierre De Sesión	32
3.10.1.3. Salir	32
3.10.2. Menú De Administración	32
3.10.2.1. Usuarios	32
3.10.2.2. Grupos.....	34
3.10.2.3. Áreas Laborales	36
3.10.2.4. Empleados	36
3.10.2.5. Listado De Empleados	38
3.10.3. Autos	39
3.10.3.1. Colores.....	39
3.10.3.2. Marcas.....	40
3.10.3.3. Modelos.....	41
3.10.3.4. Listado De Autos	42
3.10.4. Clientes/Proveedores	43
3.10.4.1. Clientes	43
3.10.4.2. Proveedores	45
3.10.4.3. Listado De Clientes/Listado De Proveedores	46

3.10.5. Inventario.....	47
3.10.5.1. Repuestos	47
3.10.5.2. Servicios.....	48
3.10.5.3. Grupos De Repuestos/Servicios	49
3.10.5.4. Lista De Precios	50
3.10.5.5. Listado De Repuestos/Servicios.....	51
3.10.5.6. Existencia De Inventario.....	52
3.10.6. Ventas	53
3.10.6.1. Aseguradoras	53
3.10.6.2. Asesores	55
3.10.6.3. Formas De Pago	56
3.10.6.4. Proforma.....	57
3.10.6.5. Orden De Trabajo.....	58
3.10.6.6. Generar Factura	59
3.10.6.7. Reportes.....	59
3.11. Paso a Paso Uso del Software.....	60
Capitulo 4	85
4.1. Mejora De Actividades Existentes	85
Capitulo 5.....	88

5.1. Conclusiones.....	88
5.2. Recomendaciones.....	88
Bibliografía	90
Anexos	91

ÍNDICE DE GRÁFICOS

3.1. Icono Instalador	25
3.2. Pantalla De Bienvenida Para Instalación.....	25
3.3. Pantalla que selecciona donde la aplicación se instalará.....	26
3.4. Pantalla para confirmar instalación	26
3.5. Instalación en Proceso	27
3.6.Instalación Completa.....	27
3.7. Icono de Acceso Directo.....	28
3.8. Mensaje De Archivo De Configuración No Creado	28
3.9. Pantalla de Configuración	29
3.10. Pantalla para ingreso.....	30
3.11. Página Principal del Sistema.....	30
3.12. Cambio de Contraseña.....	31
3.13. Usuarios	33
3.14. Miembros de los Grupos de Usuarios	33
3.15. Grupos.....	34
3.16.Permisos de Acceso para cada grupo.....	35
3.17. Datos de Áreas Laborales	36
3.18. Empleados	37

3.19. Listado de Empleados	38
3.20. Colores Autos	39
3.21. Marcas Autos	40
3.22. Modelos Autos.....	41
3.23. Listado de Autos.....	42
3.24. Clientes	43
3.25. Auto de cada cliente	44
3.26. Proveedores	45
3.27. Listado de Clientes	46
3.28. Repuestos	47
3.29. Servicios.....	48
3.30. Grupos de Repuestos/Servicios	49
3.31. Lista de Precios	50
3.32. Listado de Repuestos/Servicios	51
3.33. Existencia de Repuestos en bodega	52
3.34. Aseguradoras	53
3.35. Contactos de cada Aseguradora	54
3.36. Asesores	55
3.37. Formas de Pago.....	56

3.38. Proformas.....	57
3.39.Orden de Trabajo	58
3.40. Facturar Orden de Trabajo	59
3.41. Acceso.....	60
3.42. Inicio	60
3.43.Grupos.....	61
3.44. Crear Grupo	61
3.45.Perfiles Grupo	62
3.46.Usuarios	62
3.47. Datos Usuario.....	63
3.48. Asignación Usuario a grupo	63
3.49. Área Laboral.....	64
3.50. Creación Área Laboral.....	64
3.51. Empleados	65
3.52. Grabar Nuevos Empleados	65
3.53. Lista de Empleados	66
3.54. Colores	66
3.55. Marcas.....	67
3.56. Modelos.....	67

3.57. Ingresar datos modelos	68
3.58. Asesor	68
3.59. Aseguradoras	69
3.60. Confirmación Aseguradoras	69
3.61. Creación Cliente	70
3.62. Automóviles por cliente	70
3.63. Automóvil por cliente	71
3.64. Confirmación vehículo por cliente	71
3.65. Proveedores	72
3.66. Registro proveedores	72
3.67. Servicios / Materiales	73
3.68. Confirmación creación grupo	73
3.69. Creación sub grupo servicios	74
3.70. Creación grupo materiales	74
3.71. Registro información de repuestos	75
3.72. Registro información servicios	75
3.73. Confirmación registro información servicios	76
3.74. Generación proforma	76
3.75. Proforma manualmente	77

3.76.Confirmacion generación proforma	77
3.77.Generacion orden de trabajo.....	78
3.78.Confirmacion grabado orden de trabajo	78
3.79. Dada de baja en inventario de acuerdo a orden de trabajo.....	79
3.80. Nuevo Inventario	79
3.81. Mensaje de alerta cuando inventario no es suficiente	80
3.82. Registro condición de pago	80
3.83.Generar factura	81
3.84.Facturar a partir de OT	81
3.85.Confirmacion de grabado factura	82
3.86.Informes ordenes de trabajo.....	82
3.87. Reporte orden de trabajo.....	83
3.88. Generar Reporte de Facturas.....	83
3.89. Reporte de factura.....	84

ÍNDICE DE TABLAS

2.1. Cuadro De Las Actividades Existentes	9
2.2. Organigrama Para La Administración Del Taller	10

UNIVERSIDAD INTERNACIONAL DEL ECUADOR

FORMULARIO DE REGISTRO BIBLIOGRAFICO DE TESIS

FACULTAD DE INGENIERIA EN MECANICA AUTOMOTRIZ

ESCUELA DE MECÁNICA AUTOMOTRIZ

TÍTULO: Implementación de un Software para el control, administración y gestión
de calidad de talleres automotrices

AUTOR: Andrés Fernando Novillo Espinosa

DIRECTOR: Ing. José Andrés Castillo

ENTIDAD QUE AUSPICIO LA TESIS: N/A

FINANCIAMIENTO: SI **NO:** X **PREGADO:** X **POSGRADO:**

FECHA DE ENTREGA DE TESIS:

Día 14 Mes Septiembre Año 2010

GRADO ACADÉMICO OBTENIDO:

No. Págs. 221 **No. Ref. Bibliográfica:** **No. Anexos: 1** **No. Planos: 0**

RESUMEN:

Esta tesis de grado, abarca un estudio de las actividades de un taller automotriz, en búsqueda de administrar, controlar y mejorar los mismos.

Abarca el detalle de las actividades desde que se recibe el automóvil y se realiza la orden de trabajo, la asignación de la misma al técnico disponible, la proforma y respectiva facturación al cliente final. Incluye también actividades para la reposición de materiales necesarios para realizar todos los trabajos; incluyendo control de stocks y ubicación de los materiales dentro de la bodega.

Usamos este detalle de las actividades para desarrollar un software que facilite nuestra misión de administrar, controlar y mejorar los mismos. Este software de fácil uso y aplicación, contiene una interfase amigable, la cual ayuda a que su uso no sea un problema o una carga laboral extra, su objetivo es simplificar y ayudar en el día a día.

El objetivo de mejorar, de instituir la filosofía de la mejora continua, necesita de herramientas claras y ordenadas, las cuales podemos tener al alcance con el uso del software. Con la opción de “bajar” todo a Excel ayuda a que se pueda tabular la información para así tener la información necesaria para poder tomar decisiones en búsqueda de esta mejora.

Esta tesis significa la ayuda que puede ofrecer la informática al mundo automotriz, donde toda la experiencia, actividades, y el trabajo del día a día de todo lo que envuelve al negocio automotriz y todos los servicios que este ofrece, encontraron en

la ciencia informática a su aliado perfecto, donde encontró las herramientas necesarias para facilitar y ordenar toda la información vital para su funcionamiento.

La parte administrativa de los talleres automotrices, muchas veces olvidada o relegada a terceros, termina siendo tan importante como el conocimiento técnico, el haber detallado actividades, e implementar las herramientas necesarias para llevar un control claro, termina mostrando resultados cuantificables inmediatamente. Esta verdad significa para esta profesión el despertar hacia otras ideas, hacia un rumbo que pocas veces es tomado en cuenta, la correcta administración del taller. El valor agregado que pueden dar estas herramientas de control es inmenso, y comprender esto es el primer paso hacia la mejor continúa.

Instituir más y mejores actividades es vital para el éxito de todas las organizaciones, buscar siempre ayuda en las herramientas tecnológicas.

Entender todas las actividades inmersas en entregar un servicio de calidad, y además optimizar las mismas para que el margen de utilidad sea mayor.

Ampliar el campo de aplicación de estas herramientas dentro de los servicios automotrices.

PALABRAS CLAVES:

- Software
- Taller
- Automotriz
- Proceso

- Estandarización
- Administración
- Trabajo
- Control

MATERIA PRINCIPAL: 1. Administración y organización de talleres

- MATERIA SECUNDARIA:**
1. Gerencia de servicios
 2. Administración por calidad total
 3. Programación del mantenimiento

TRADUCCIÓN AL INGLES

TITLE: Implementation of Software for monitoring, administration and management of automotive shops

ABSTRACT:

This thesis covers a study of the activities of an auto shop, seeking to manage, monitor and improve them.

It covers the details of the activities from the reception of the car and making the work order, assigning it to the available technician, making the respective quotation and final customer billing. It also includes activities for the replenishment of materials needed to complete all assignments, including stock control and location of materials within the warehouse.

We use this detailed activities to develop software that facilitates our mission to manage, monitor and improve them. This user-friendly software and application contains a user-friendly interface, which helps their use is not a problem or an extra workload, the goal is to simplify and assist in day to day. The aim to improve, to institute the philosophy of continuous improvement requires clear and neat tools, which we have within reach using the software. With the option to "download" everything Excel can help to tabulate the data in order to have the

necessary information to make decisions in pursuit of this improvement. This thesis proves the help that informatics provides to the automotive world, where all the experience, activities, and day-to-day with everything that surrounds the automobile business and all the services it offers, found in computer science a perfect partner, where it found the necessary tools to facilitate and manage vital information for its functioning.

The administrative part of auto repair shops, often forgotten or relegated to third, ends up being as important as technical knowledge, have detailed activities, and implement the necessary tools to track course, ends up showing measurable results immediately. This truly means to wake up this profession to other ideas, to a course that is rarely taken into account, the proper administration of the workshop. The added value that can provide these tools of control is huge, and understanding this is the first step towards continuous improvement

Instituting more and better activities is vital to the success of all organizations, always seeking help in technological tools.

Understand all the activities embedded in delivering a quality service, and also optimize them so that the profit margin is higher

Expand the scope of these informatics tools within the automotive services.

KEYWORDS:

- Software
- Workshop
- Automotive
- Process
- Standardization
- Administration
- Labour
- Control

FIRMAS:

.....

DIRECTOR

.....

GRADUADO

INTRODUCCIÓN

En el día a día de un taller de mecánica automotriz es muy habitual estar mayormente inclinado hacia la reparación de los vehículos que hacia la propia administración del taller; sin embargo, la administración es algo de lo que difícilmente los talleres pueden librarse. Por ello, es necesario tomar en cuenta a cuánto ascienden los costos de una empresa al tratarse de las largas horas que toman en hacer cuentas, en conocer los ingresos y egresos del taller, en sacar reportes con el estatus del auto dentro del proceso de revisión o reparación, en preparar la información para la contabilidad, en imprimir facturas, en realizar los pagos a proveedores, en fin, la lista crece con respecto a las actividades administrativas, así como el tiempo que esto toma y el costo que implica para el taller.

Cuando la administración de un taller se conserva de la forma tradicional, el taller tiene la asistencia de una persona para cumplir con las funciones de esta área; es decir, esta persona será la responsable de la administración, será quien realice la documentación administrativa del taller, y será quien realmente conozca y maneje los detalles de la parte administrativa. Aparentemente esto es ventajoso y fácil puesto que se está dando la responsabilidad administrativa a otra persona, quien parece lo maneja bien, a más de que permite que el dueño no tenga la necesidad de involucrarse directamente en esa parte del negocio, pero la verdad de esto es que no es ventajoso, ya que el que el buen funcionamiento del taller depende únicamente

del conocimiento y la administración de una persona; además, de que no hay un control al respecto.

La transición entre la forma anterior de administrar hacia el uso del software, en primera instancia será complicada, debido a la cantidad de información que se tendrá que comenzar a ordenar.

Afortunadamente, existen algunas herramientas que nos ayudarán en la administración de talleres mecánicos, que al igual que las que utilizamos para reparar los vehículos, nos hacen la vida mucho más sencilla, ese tipo de herramienta es el software.

El presente proyecto consiste en la implementación de un Software para la Administración y gestión de calidad de talleres automotrices, que es quizás hoy en día una de las herramientas más utilizadas.

Con el programa de computación o software para talleres mecánicos automotrices, cada actividad en la administración se hará siempre de la misma forma y no dependerá más del criterio de la personas. La información siempre estará en un solo lugar independientemente de quien la genere o quien la use, es decir que la perdida de información ya no será un problema. La obtención de información será inmediata y siempre estará disponible. No importa si la persona que maneje el programa renuncia, la siguiente, hará las cosas de la misma forma que las hacia la anterior, porque así lo requerirá el programa.

Cada día más talleres mecánicos, están adoptando esta tecnología para estar a la altura de las exigencias que el mercado establece, haciendo su trabajo más simple, más eficiente y por supuesto incrementando sus ingresos.

CAPÍTULO 1

1.1. MARCO TEÓRICO

1.1.1 Concepto de calidad.

Calidad significa, entregar un producto o servicio, minimizando todos los factores que puedan significar un cierto grado de inconformidad con nuestros clientes internos y externos

1.1.2 Historia de la calidad.

El concepto de calidad no es algo nuevo, se conoce de que culturas de siglos pasados ya tenían, o buscaban siempre se entreguen productos o servicios de alta calidad, es así como El **Código de Hammurabi** (1752 a. C.), declaraba: “Si un **albañil** construye una casa para un hombre, y su trabajo no es fuerte y la casa se derrumba matando a su dueño, el albañil será condenado a muerte”.

Más adelante los artesanos eran los controladores de calidad, ellos eran a la vez instructores e inspectores de calidad de sus aprendices, ya que ellos conocían el producto, el proceso y sus clientes. Al llegar la revolución industrial y el inicio de fabricación en línea, el departamento de producción era el responsable de la calidad, pero se priorizó más la producción sobre la calidad, lo cual afectaba directamente a la calidad, para lo cual las direcciones crearon departamentos específicos para el control de la calidad

Luego de la segunda guerra mundial, los japoneses buscaron la forma para cambiar la percepción que se tenía mundialmente acerca de sus productos,

ellos tomaron a la calidad como una cultura en la cual están envueltos y son partes activas y responsables todos y cada uno de los miembros de la organización.

1.1.3 Herramientas estadísticas para la calidad

Para el control de la calidad, se necesita también no ser reactivo frente a los problemas de baja calidad, se tiene que instituir una cultura de pro actividad, de planificación y control para todas las actividades, para esto existen herramientas estadísticas que nos ayudaran a tomar decisiones.

Las herramientas son las siguientes

1.1.3.1 Hoja de control

El objetivo primordial de este llevar un registro del fenómeno que se quiere estudiar del servicio o material. Lo importante de esta herramienta es que los datos puedan ser cualitativos o cuantitativos. Es decisión de la organización la forma como se estudiara esta información

1.1.3.2 Histograma

Es la presentación de medidas clasificadas y ordenadas, donde se fijan valores máximos y mínimos, contra los cuales podemos definir el rango de las series de las medidas. Uno de sus objetivos será mostrar el resultado de un cambio en el sistema e identificar anomalías examinando la forma.

1.1.3.3 Diagrama de Pareto

Esta herramienta nos sirve para priorizar los problemas o las causas que generan un problema. Se basa en la teoría del economista italiano

VILFREDO PARETO, cuya teoría era: si se tiene un problema con muchas causas, podemos decir que el 20% de las causas resuelven el 80 % del problema y el 80 % de las causas solo resuelven el 20 % del problema.

1.1.3.4 Diagrama de causa – efecto

Su función es un poco más apegada hacia lo educativo, es para que la gente comprenda su función e importancia dentro del proceso o procesos que se necesitan para entregar un producto final.

1.1.3.5 Diagramas de flujo

Diagrama de Flujo es una representación gráfica de la secuencia de etapas, operaciones, movimientos, decisiones y otros eventos que ocurren en un proceso. Esta representación se efectúa a través de formas símbolos y gráficos.

1.1.3.6 Diagramas de dispersión

Estos gráficos nos ayudan a buscar la correlación positiva o negativa entre dos variables

1.1.4 Administración por calidad total.

La administración por calidad total es un concepto que en base a sistemas y a al involucramiento de todos los miembros de una organización, busca elevar conscientemente la calidad de sus procesos, productos y servicios, y a la vez tener esta mejora a lo largo del tiempo.

Para esto podemos citar 4 elementos que nos ayudaran a mantener esta filosofía de mejora continua en toda la organización

- Un método de control.- PLANEAR – HACER – VERIFICAR – ACTUAR

- Procesos estadísticos.- no se puede controlar aquello que no se mide
- Trabajo en equipo
- Capacitación

1.1.5 Mejoramiento continuo.

Es incrementar la probabilidad de aumentar la satisfacción de los clientes, para lo cual se debe establecer objetivos claros y medibles, se deben evaluar estos objetivos, luego de evaluados se debe proponer soluciones y se debe controlar la implementación de esta solución.

Los resultados se revisaran de acuerdo a pautas o a direcciones de los altos mandos de la empresa, de esta manera y cumpliendo con todo lo mencionado estaremos encaminados hacia procesos de mejora continua.

1.1.6. Gestión de la calidad

La gestión de la calidad es el conjunto de normas dentro de una organización, estas normas deben estar interrelacionadas entre si, y con esto se administra la calidad dentro de la empresa en busca de la mejor continua.

Se debe definir dentro de la organización claramente los siguientes aspectos

- Estructura de la organización
- Estructura de responsabilidades
- Procedimientos
- Procesos
- Recursos

La gestión de la calidad de una empresa puede ser validada y certificada por organismos normalizadores, ejemplo de eso es la certificación ISO 9001

1.1.7. Software como herramienta de control de calidad.

El software, ayuda a las organizaciones a que se origine, se procese y se almacene información de acuerdo a las actividades, y objetivos planteados por la dirección.

El orden y el fácil acceso a la información es de vital importancia para realizar las evaluaciones necesarias en búsqueda de soluciones y mantener la política de mejora continua.

1.1.8. Visual Studio.Net

Microsoft .NET es una plataforma de desarrollo y ejecución de aplicaciones. Esto quiere decir que no sólo nos brinda todas las herramientas y servicios que se necesitan para desarrollar modernas aplicaciones empresariales y de misión crítica, sino que también nos provee de mecanismos robustos, seguros y eficientes para asegurar que la ejecución de las mismas sea óptima. Los componentes principales de la plataforma .NET son:

- Un entorno de ejecución de aplicaciones, también llamado “Runtime”, que es un componente de software cuya función es la de ejecutar las aplicaciones .NET e interactuar con el sistema operativo ofreciendo sus servicios y recursos.

- Un conjunto de bibliotecas de funcionalidades y controles reutilizables, con una enorme cantidad de componentes ya programados listos para ser consumidos por otras aplicaciones.
- Un conjunto de lenguajes de programación de alto nivel, junto con sus compiladores y linkers, que permitirán el desarrollo de aplicaciones sobre la plataforma .NET.
- Un conjunto de utilitarios y herramientas de desarrollo para simplificar las tareas más comunes del proceso de desarrollo de aplicaciones
- Documentación y guías de arquitectura, que describen las mejores prácticas de diseño, organización, desarrollo, prueba e instalación de aplicaciones .NET

1.1.9. ¿Qué es Visual Basic .NET

Es un lenguaje orientado a objetos, formando parte de Visual Studio y compartiendo el entorno de desarrollo con Microsoft Visual C++ .NET, Microsoft Visual C# .NET, etc.

1.1.10. Entorno de Desarrollo

El Entorno de Desarrollo recibe el nombre de Entorno de Desarrollo de Microsoft Visual Studio .NET. Este entorno es personalizable y contiene todas las herramientas, necesarias para construir programas para Microsoft Windows.

El Entorno de Desarrollo contiene múltiples ventanas y múltiples funcionalidades y es por consecuencia llamado un entorno de desarrollo integrado (Integrated Development Environment IDE).

CAPÍTULO 2

Estudio situación actual y propuesta de perfiles laborales

En estos tiempos se ha facilitado la administración de los talleres de reparación automotriz medianos y pequeños, con la ayuda de los programas de computación diseñados especialmente para este propósito, estas herramientas permiten administrar de manera muy fácil y eficiente todas las actividades relacionadas con los servicios que se realizan en un taller, de esta manera se eliminan los tiempos que se invierten en tareas tediosas y dedicándolo a actividades verdaderamente productivas.

El rango de programas para administrar y gestionar talleres automotrices es muy amplio, por esta razón a la hora de diseñar un programa para tal función, se debe poner mucha atención en los aspectos del taller tales como tamaño del taller, actividades que en se desarrollan; es decir los puntos más importantes sobre los que se trabajará con el Software para el taller Automotriz.

“Si comprometemos los programas de computación, corremos el riesgo de perder habilidades básicas de largo plazo. Si fallamos en tomar en cuenta las preocupaciones de los profesionales del área, corremos el riesgo de quedar obsoletos. La clave es alcanzar el balance correcto, pero hay más de una manera de lograrlo.”¹

¹ FREEMAN , Peter. Elements of Effective Computer Science. IEEE Computer, vol. 30, no. 11, p. 47. Nov. 1997

En esta época de la humanidad el trabajo mental adquiere muchísima importancia frente a la revolución informática así como el trabajo físico lo fue en su momento para la revolución industrial. Esto actualmente ha ocasionado que la sociedad dependa de un recurso que nos proveen los programas de computación: La información renovable, que nos sirve para la toma de decisiones.

Ese es uno de los objetivos principales de la universidad y ha sido también una motivación personal importante para la realización de este trabajo.

2.1. DESGLOSE DE ACTIVIDADES EXISTENTES

Actividad	Area responsable	Descripcion
Recepcion vehiculos	Administracion	Se llena un formato de recepcion de vehiculo
Entrega de vehiculos	Administracion	El auto sale de taller entregando la hoja de entrada
Solicitud de materiales	Bodega	El personal de bodega solicita material, cuando el anterior se acaba o esta a punto de acabarse, no se consideran tiempos de entrega
Compra materiales	Administracion	Compra materiales en base a solicitud de bodega

Tabla 2.1. Cuadro de las Actividades Existentes

2.2. PLANTEAMIENTO DE NUEVO ORGANIGRAMA PARA ADMINISTRACION TALLER

El siguiente es el organigrama propuesto para la administración del taller

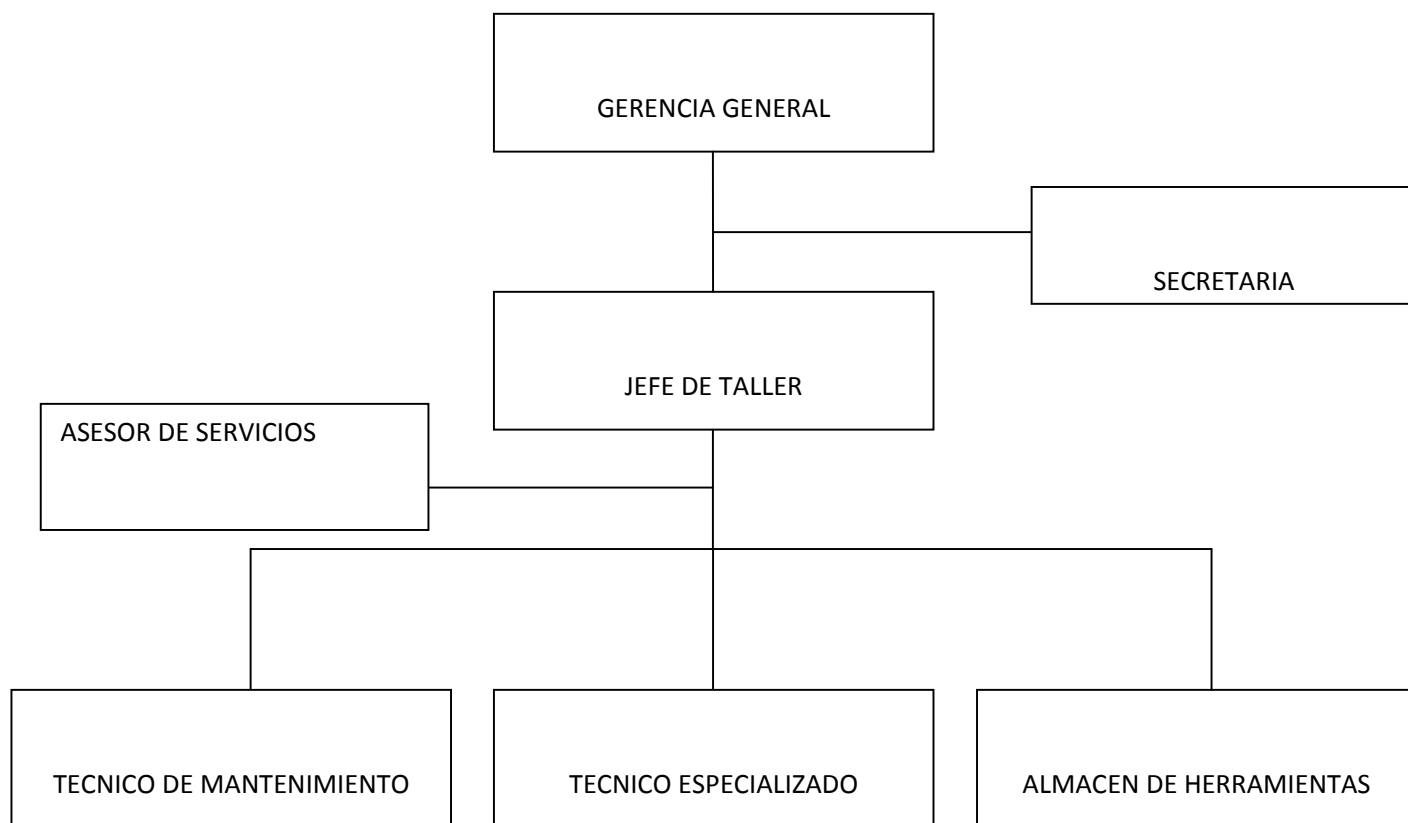


Tabla 2.2. Organigrama para la administración del Taller

2.3. PROPUESTA PERFILES LABORALES

2.3.1. Detalle de las funciones de cada responsable

2.3.1.1. Jefe de taller mecánico

- Finalidades principales:
 - a.- Administrar los recursos humanos y equipo con que cuenta el taller mecánico para el desempeño de sus actividades.
 - b.- Desarrollar planes de trabajo para el funcionamiento del taller mecánico
 - c.- Evaluar el desempeño de los técnicos para determinar su nivel de eficiencia en el desempeño de su trabajo
 - d.- Supervisar las actividades de los técnicos para asegurar que se cumpla con los requerimientos de tiempo y calidad en sus trabajos.
 - e.- Asignar órdenes de trabajo para su ejecución.
 - f.- Realizar controles de calidad durante la ejecución de los trabajos.
 - g.- Realizar inspecciones previa entrega de servicios terminados.
 - h. Desarrollar planes de entrenamiento y capacitación para el personal.

- i. Fomentar e instituir la cultura de la mejora continua.
 - j. Definir objetivos claros y medibles para las actividades del taller.
 - k. Verificar que se cumplan con las normas y procedimientos de seguridad e higiene industrial
- Objetivo general: Administrar y coordinar los recursos humanos y equipo asignados al taller mecánico así como todas las actividades del mismo; todas ellas apegadas a los lineamientos y metas fijadas por el taller y a los requerimientos de tiempo y calidad estipuladas por el solicitante
 - Principales relaciones internas que el titular del puesto debe desarrollar y mantener, para lograr sus resultados:
 - a.- Recursos humanos para resolución de asuntos laborales relacionados con el personal que labora en esta área.
 - b.- Todas las coordinaciones del taller para recepción de información
 - c.- Principales relaciones externas que el titular del puesto debe desarrollar y mantener, para lograr sus resultados
 - d.- Proveedores de materiales para obtener cotizaciones de materiales y equipo para el taller mecánico.

e.- Proveedores de servicios para obtener cotizaciones de servicios solicitados.

2.3.1.2. Asesor de servicio

- Finalidades principales:

a.- Adquirir herramientas y materiales

b.- Seleccionar herramienta y equipo

c.- Elaborar periódicamente reportes sobre el desarrollo de las actividades para el control de las mismas

d.- Optimizar los suministros tecnológicos

e.- Recepción de vehículos y realizar ordenes de trabajo

- Objetivo general: Coordinar el frecuente abastecimiento de herramientas y equipos para el taller, y realizar reportes de todas las órdenes de trabajo realizadas. Distribución de carga laborar a todos los técnicos

- Principales relaciones internas que el titular del puesto debe desarrollar y mantener, para lograr sus resultados:

a.- Objetividad para el desarrollo de informes de personal, y organización para el continuo suministro de materiales

- Principales relaciones externas que el titular del puesto debe desarrollar y mantener, para lograr sus resultados:

a.- Proveedores para determinar las mejores condiciones del mercado para la adquisición de materiales y equipos.

2.3.1.3. Coordinador de Mantenimiento

- Finalidades principales:

a.- Dar mantenimiento a equipo instalado en el taller.

- Objetivo general: Crear planes de mantenimiento preventivos y correctivos para todos los equipos del taller.

- Principales relaciones internas que el titular del puesto debe desarrollar y mantener, para lograr sus resultados:

a.- Comunicación continua con todos los usuarios de los equipos, para analizar posibles fallas que estén ocurriendo en los equipos.

b.- Organización y planificación con el jefe de taller para la compra de partes y repuestos para los mantenimientos de los equipos.

c.- Designación de subalternos.

2.3.1.4. Técnico Especializado

- Finalidades principales:

a.-Realizar reparaciones y mantenimientos del automóvil de acuerdo a lo indicado en la orden de trabajo

- Objetivo general: Dar servicio de reparación y mantenimiento de automóviles acorde a lo estipulado en la orden de trabajo y según recomendaciones de sus superiores.
- Principales relaciones internas que el titular del puesto debe desarrollar y mantener, para lograr sus resultados:

a.-Clientes para corroborar parámetros del trabajo.

2.3.1.5. Técnico de Mantenimiento

- Finalidades principales:

a.- Dar mantenimiento a los equipos y herramientas para satisfacer las necesidades de operación del taller.

- Objetivo general: Dar mantenimiento a las estructuras metálicas, equipos y herramientas del taller.

2.3.1.6. Almacén de Herramientas

- Finalidades principales:
 - a.- Llevar el control de entrada y salida de materiales herramientas y consumibles.
 - b.- Elaborar vales de préstamo de herramientas con tiempo limite de entrega.
 - c.- Mantener informado a la jefatura de las necesidades de herramientas o consumibles que se generen en el proceso de compra
 - d.- Hacer los pedidos a la jefatura de herramientas y consumibles en base a un mínimo y un máximo de existencias
- Objetivo general: Mantener el control de existencias de herramientas y consumibles dentro del almacén y zonas de trabajo.
- Principales relaciones internas que el titular del puesto debe desarrollar y mantener, para lograr sus resultados:
 - a.- Personal del taller: Mecánico, para dotarlos de materiales y herramientas

CAPÍTULO 3

Propuesta de nuevas actividades e implementación del software

3.1. TIPO DE EMPRESA A LA QUE SE APLICA EL PROGRAMA

Generalmente los programas para administración de talleres no solo pueden ser utilizados por talleres mecánicos, debido a que la actividad que realizan es muy similar, estos programas pueden ser utilizados por talleres de reparación de sistema eléctrico, especializados en frenos, transmisiones automáticas, área acondicionado, hojalatería y pintura, de reparación de motocicletas entre otros.

3.2. INFORMACIÓN PARA EL REGISTRO DE ÓRDENES DE TRABAJO

De acuerdo al nivel de detalle que se requiera registrar para cada orden de trabajo, el programa deberá como mínimo registrar las placas, el vehículo, el nombre del cliente, su dirección, su teléfono, además deberá registrar los repuestos que el vehículo utilizara, si provienen de un almacén externo o se consumen de almacén propio, debe poder registrar observaciones importantes de situaciones que se pudieran presentar durante la estancia del vehículo en el taller , también es importante que pueda registrar gastos adicionales relacionados con el trabajo, costos y precios de todo lo que se le hace a cada orden de trabajo.

3.3. MANEJO DEL CATÁLOGO DE UN CATÁLOGO DE CLIENTES

Es muy importante que el software cuente con un manejo de catalogo de clientes, ya que es ahí donde se tiene toda la información identificación. El programa debe proporcionar mecanismos de búsqueda de información por múltiples datos como puede ser por nombre, apellidos, por el número de teléfono, por la dirección o por segmentos de cualquiera de estos datos.

3.4. MANEJO DE INVENTARIO DE REPUESTOS

Debido a que algunos talleres manejan un inventario de repuestos propio, si este es el caso, es importante que el programa seleccionado cuente con un módulo que permita controlar las entradas y las salidas de repuestos por los diferentes conceptos, como pueden ser consumos de repuestos por ordenes de trabajo, ingreso de mercancía nueva, ajustes de entrada y ajustes de salida, es importante que se puedan definir varios almacenes y localizaciones dentro de estos para un adecuado manejo de los artículos. Es importante también que permita hacer inventarios físicos para poder mantener la información al día.

3.5. MANEJO DE REPORTE

Ya que uno de los objetivos del programa es ir registrando la información conforme transcurre el tiempo, es importante que el programa cuente con una serie de reportes que permitan estar informado en cualquier momento de los indicadores mas importantes de taller, como son el importe de los trabajos pendientes de pago

para cuando se maneja crédito, el consumo de repuestos, consumo de repuestos por vehículo, ingresos por mecánico, inventario físico, venta de repuestos a público en general, comparativo ingresos/ gastos por rango de fechas, facturación e impresión de contra recibos entre otros .

3.6. SEGURIDAD DE LA INFORMACIÓN QUE SE MANEJA

Otro de los factores importantes en un programa para gestionar un taller mecánico es el que respecta a la seguridad, se debe verificar si el programa puede ser utilizado en una sola computadora o puede trabajar en una red donde lo utilizan varios usuarios al mismo tiempo, si este es el caso, debe contar con un mecanismo de seguridad donde a cada persona que lo utilice se le asigne un usuario y una contraseña para su identificación dentro del sistema, y donde a cada una se le asignen permisos de acceso solo a las áreas del programa que su trabajo requiera, además dentro de cada área de trabajo es importante que se pueda definir que acciones puede realizar , por ejemplo una secretaria se le pueden asignar permisos para captura de información pero no se le debe permitir borrarla, de esta forma se puede garantizar que cada usuario solo pueda realizar solo las funciones que le correspondan.

3.7. IDENTIFICACIÓN DE LAS ACTIVIDADES ADMINISTRATIVAS

a.- Administración de los recursos humanos y equipo con que cuenta el taller mecánico para el desempeño de sus actividades.

- b.- Desarrollo de planes de trabajo para el funcionamiento del taller mecánico
- c.- Supervisión de las actividades de los técnicos para asegurar que se cumpla con los requerimientos de tiempo y calidad en sus trabajos.
- d.- Asignación de órdenes de trabajo para su ejecución.
- e.- Adquisición de herramientas y materiales
- f.- Elaboración periódica de reportes sobre el desarrollo de las actividades para el control de las mismas
- g.- Optimización de los suministros tecnológicos
- h.- Mantenimiento a equipo instalado en el taller
- i.- Mantenimiento al equipo mecánico de los laboratorios y talleres.
- j.- Control de entrada y salida de materiales herramientas y consumibles
- k.- Informar a la jefatura de las necesidades de herramientas o consumibles que se generen en el proceso de manufactura
- l.- Hacer los pedidos a la jefatura de herramientas y consumibles en base a un mínimo y un máximo de existencias

3.8. CLASIFICACIÓN Y ORGANIZACIÓN DE ACTIVIDADES

3.8.1 Actividades administrativas de trabajo en el taller

Descripción de cada una de las etapas de las actividades administrativas en el taller:

3.8.1.1. Atención a Órdenes de trabajo:

- Necesidad de realización de un trabajo: Toda necesidad referente al mantenimiento un vehículo, será atendida a través de una orden de trabajo.
- Análisis de la información o detección de la necesidad: Este es uno de las actividades más importantes, ya que dependiendo del buen entendimiento y análisis de la información, será el éxito del trabajo final; Por lo tanto primero se analiza la petición y se cuestiona tantas veces como sea necesario para dejar la idea bien clara y entendida.
- Replantea la propuesta en base a la información obtenida: Dependiendo de la necesidad es importante dedicarle el tiempo necesario para captar bien la propuesta, por lo tanto en algunas ocasiones es necesario tener de 2 a 3 reuniones antes de iniciar cualquier otra actividad, para lo cual se deben hacer siempre las observaciones pertinentes.
- Análisis del replanteamiento o la propuesta de solución: El cliente analiza la Información buscando el aprobar la información recibida.
- Planeación de cargas de trabajo y de existencia de materiales: Dependiendo del número de solicitudes y de la carga de trabajo interno, así como de los materiales se planea el tiempo de entrega del trabajo

- Elaboración de la orden de trabajo y de la solicitud de materiales: Se realiza el levantamiento oficial a través de la Orden de trabajo, donde se especifican las características del trabajo a realizar. Además se solicita los materiales necesarios para la elaboración del trabajo.

3.8.1.2. Asignación de trabajo:

El trabajo se asigna a un trabajador en función de su carga de trabajo y este una vez que recibe los materiales empieza a ejecutar la orden de trabajo.

3.8.1.3. Desarrollo y ejecución de la orden de trabajo:

Se procede a la ejecución del trabajo, empleando la infraestructura instalada.

3.8.1.4. Actividad terminada:

Una vez terminado el trabajo, se entrega al jefe del taller para que de su visto bueno, o en su caso solicite las correcciones adecuadas. Por otro lado una vez que el jefe del taller dio su visto bueno llama al cliente para hacerle entrega del trabajo, el cual debe evaluar el trabajo y dar su visto bueno de aceptación del trabajo terminado, ya que de otra forma se regresa el automóvil a ser reprocesado.

3.8.1.5. Archiva información y elabora reportes mensuales:

Una vez terminado el trabajo, el cliente firma la orden de trabajo dando su aceptación del mismo y se procede a archivar el mismo para que al fin de la semana se genere la información del reporte trimestral de trabajos realizados.

3.8.1.6. Analiza reportes trimestrales para la toma de decisiones:

El Jefe de taller recibe los reportes trimestrales, lo que le permite informar a sus superiores y a la vez poder tomar decisiones estratégicas respecto a la dirección que debe seguir el taller mecánico.

3.8.2. Descripción de cada una de las etapas de compras de herramientas, materiales y consumibles

3.8.2.1. Solicitud de Compra ya sea de Materiales, Herramientas o consumibles:

Esta solicitud se llena por un lado en base a necesidades detectadas por el personal encargado del almacén de herramientas, materiales y consumibles y por otro lado en base al plan de mantenimiento preventivo de la maquinaria y equipos del taller mecánico.

- Solicitud: Para el caso de las herramientas se consiguen 3 cotizaciones de diferentes proveedores y para los consumibles solo se realiza una lista y se estima el costo de los mismos.

3.8.2.2. Evaluación de las peticiones para consumibles y de las cotizaciones:

El Jefe del Taller evalúa las cotizaciones y los costos de consumibles y ya sea que autorice la compra de los mismos o que solicite nuevas cotizaciones o en su caso la reducción de las cosas a comprar.

3.8.2.3. Etapa administrativa para la adquisición de las herramientas, materiales y consumibles:

El personal administrativo designado se encarga de adquirir los mismos.

3.8.2.4. Recepción del producto:

El jefe del taller mecánico recibe el producto y si este cumple con las especificaciones solicitadas firma la factura dando su visto bueno y entrega la misma a la secretaria, y entregando el producto al almacén interno con lo cual se cierra la actividad.

3.9. IMPLEMENTACION DEL SOFTWARE

El sistema es de instalación sencilla y amigable para el usuario. Los pasos siguientes permiten una correcta instalación del sistema.

- 1) El primer paso para la instalación del sistema es hacer doble click en el icono instalador del sistema.



Gráfico 3.1. Icono instalador

2) Al momento de realizar el paso 1 aparecerá la siguiente pantalla.

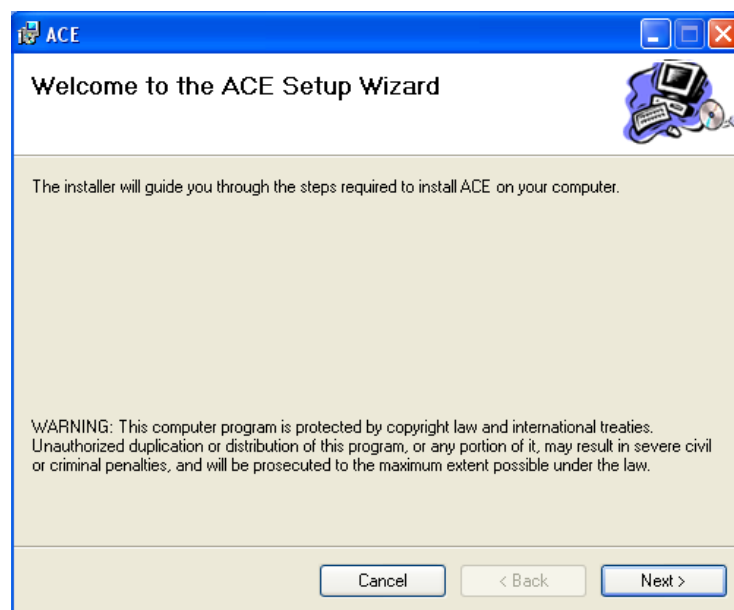


Gráfico 3.2. Pantalla de Bienvenida para instalación

3) Hacer click en el botón siguiente.

4) La siguiente pantalla indicará el path donde se quiere que la aplicación se instale; también pedirá elegir si el sistema a instalar será usado solo por el usuario respectivo de Windows o para todos aquellos que usen la PC. En este caso se elegirá "TODOS".

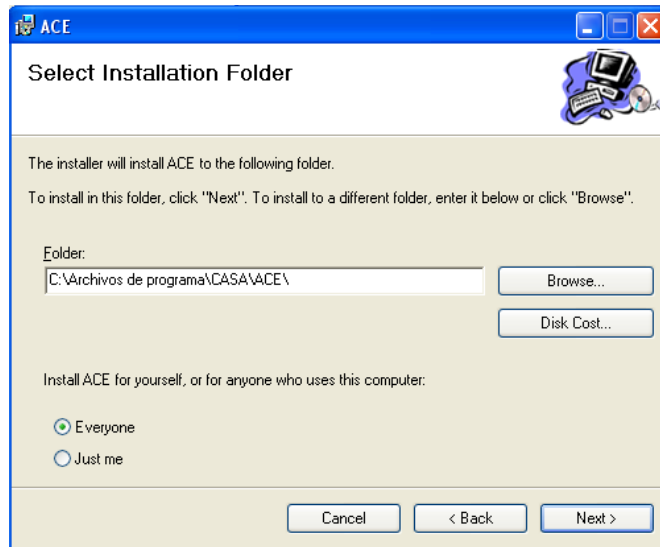


Gráfico 3.3. Pantalla que selecciona donde la aplicación se instalará

5) La siguiente pantalla es solo para confirmar la instalación, se elegirá siguiente.

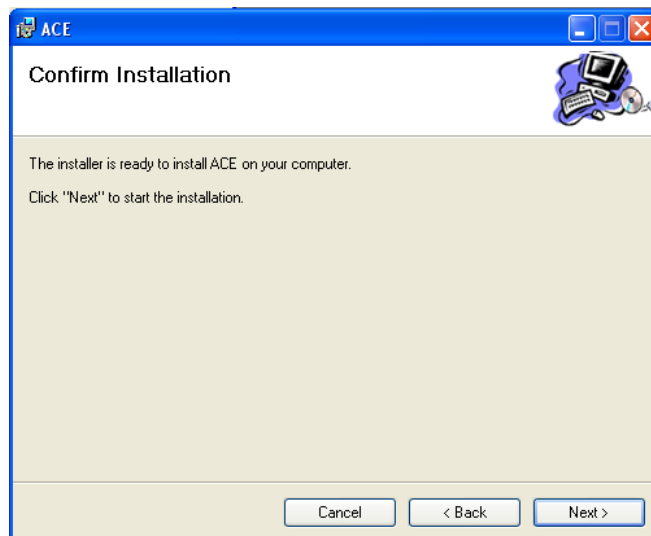


Gráfico 3.4. Pantalla para confirmar instalación

6) La aplicación se instalará con todos sus componentes y requerimientos.

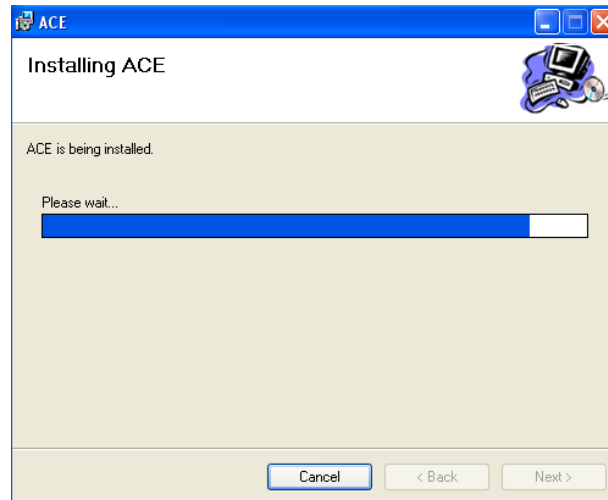


Gráfico 3.5. Instalación en proceso

- 7) La siguiente pantalla indica que el sistema se ha instalado con éxito, cerramos la aplicación.

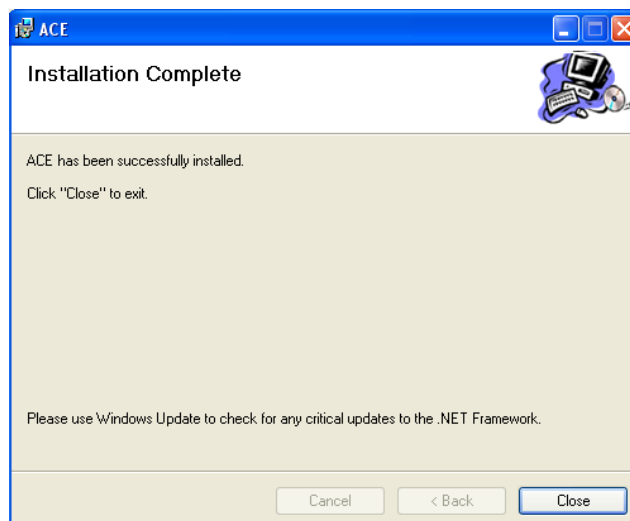


Gráfico 3.6. Instalación completa

- 8) Se creará un icono (acceso directo) tanto en el escritorio, como también en inicio, programas, ACE, indicando que la aplicación se ha instalado satisfactoriamente.



Gráfico 3.7. Icono de acceso directo

3.10. FUNCIONAMIENTO DEL SISTEMA

El sistema A.C.E. permite tener un control sobre los empleados, y de esta de manera poder realizar un correcto manejo tanto de recursos como de personal.

Al momento de instalar el sistema se crea un acceso directo en el escritorio con el cual se tiene un rápido acceso a la aplicación. Al hacer doble click en el icono de acceso directo, el sistema buscará el archivo de configuración del sistema, en caso de que no exista dicho archivo se desplegará el siguiente mensaje.

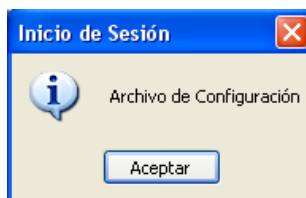


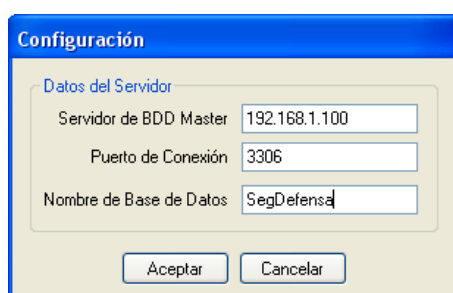
Gráfico 3.8. Mensaje de archivo de configuración no creado

Este mensaje nos indica que el archivo de configuración para la conexión al servidor no se encuentra creado. Al realizar click en el botón de “Aceptar” aparecerá la ventana de configuración.

Esta ventana posee 3 opciones de configuración, los cuáles son:

- **Servidor de BDD Master:** Esta opción es la que recibirá la dirección IP del servidor (lugar donde se encuentra instalada la Base de Datos).
- **Puerto de Conexión:** Debido a que se utilizará el motor de Base de Datos MySQL, es necesario ingresar el puerto predeterminado 3306.
- **Nombre de la Base de Datos:** Se ingresará el nombre de la Base donde se encuentra toda la información. El nombre predeterminado es SegDefensa.

Nota: En caso de que se realice algún cambio en la configuración del servidor, se debe configurar nuevamente la pantalla de configuración, oprimiendo F2 mientras se tiene el enfoque en el usuario.



The image shows a Windows-style dialog box titled "Configuración". It contains a section labeled "Datos del Servidor" with three text input fields. The first field is "Servidor de BDD Master" containing "192.168.1.100". The second field is "Puerto de Conexión" containing "3306". The third field is "Nombre de Base de Datos" containing "SegDefensa". At the bottom of the dialog are two buttons: "Aceptar" and "Cancelar".

Gráfico 3.9. Pantalla de configuración

Una vez realizada las configuraciones correctamente, se presentará la pantalla de inicio de sesión. Esta pantalla pide en nombre de usuario y contraseña (cada usuario y contraseña se le otorgará a cada empleado de manera individual).

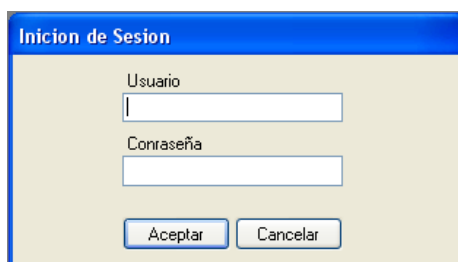


Gráfico 3.10. Pantalla para ingreso

Una vez ingresado al sistema, con un usuario y una contraseña correcta, se ingresa al sistema y se tendrá acceso a la página principal, la cual posee el menú para el manejo del sistema. De acuerdo al perfil de usuario se crearán las opciones del menú.

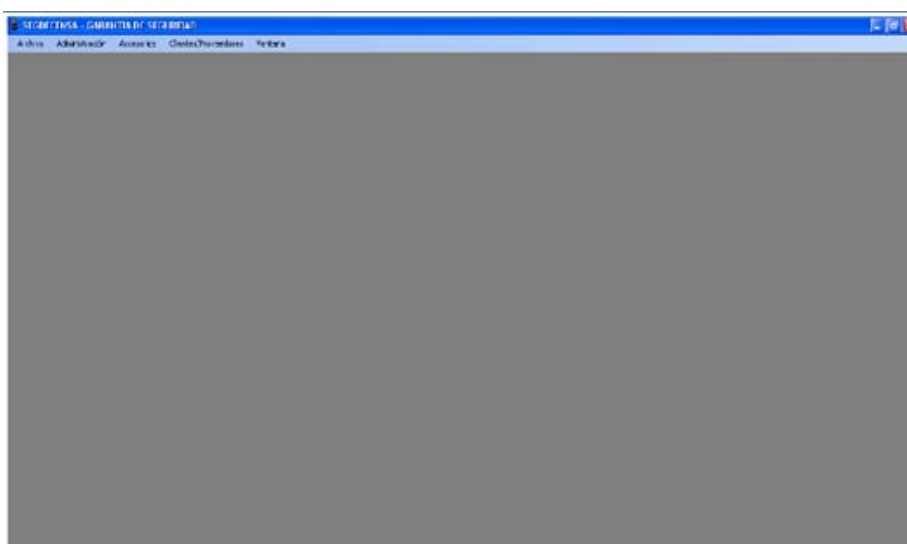


Gráfico 3.11. Página principal del sistema

3.10.1. Menú De Archivo


La opción Archivo es la que controla el funcionamiento general del sistema. En esta opción se encontrará todo lo referente al cambio de contraseña, cierre de sesión y salida del sistema.

3.10.1.1. Cambiar Contraseña

Esta opción permite al usuario, como su nombre lo indica, cambiar de contraseña en caso de pérdida, seguridad o simplemente deseo.

Los requerimientos de esta pantalla son las siguientes:

- **Antigua Contraseña:** Es donde se ingresará la contraseña que se desea cambiar.
- **Nueva Contraseña:** Es la contraseña que reemplazará, o por la cual se cambiara la antigua contraseña.
- **Confirmar Contraseña:** Pide el reingreso de la nueva contraseña para confirmar el cambio.



The image shows a screenshot of a software dialog box titled "Cambio de Clave". The dialog has a blue header bar with the title and standard window control icons (minimize, maximize, close). Below the header, there is a section labeled "Datos" which contains three text input fields. The first field is labeled "Antigua Contraseña", the second is "Nueva Contraseña", and the third is "Confirmar Contraseña".

Gráfico 3.12. Cambio de Contraseña

3.10.1.2. Cierre de Sesión

Este submenú permite al usuario regresar a la pantalla original, en la cual se ingresan el usuario y la contraseña.

3.10.1.3. Salir

Cierra y sale el usuario del sistema.

3.10.2. Menú De Administración

3.10.2.1. Usuarios

Este submenú se divide en 2 secciones. En la sección de usuarios es donde se creará los registros de las personas que utilizarán el sistema. En la sección de miembros de grupos es donde se asignará a cada usuario el grupo al que pertenecerá. Cada grupo posee cierto nivel de acceso dentro del sistema.

En la sección de Usuarios se debe ingresar los datos de la persona que utilizará el sistema. La opción deshabilitado sirve para denegar, o como su nombre lo indica, deshabilitar el uso del sistema para dicho usuario.

Existe un pequeño recuadro que nos muestra un listado de todos los usuarios del sistema. Al hacer doble click en cualquier usuario podemos recuperar los datos del mismo.

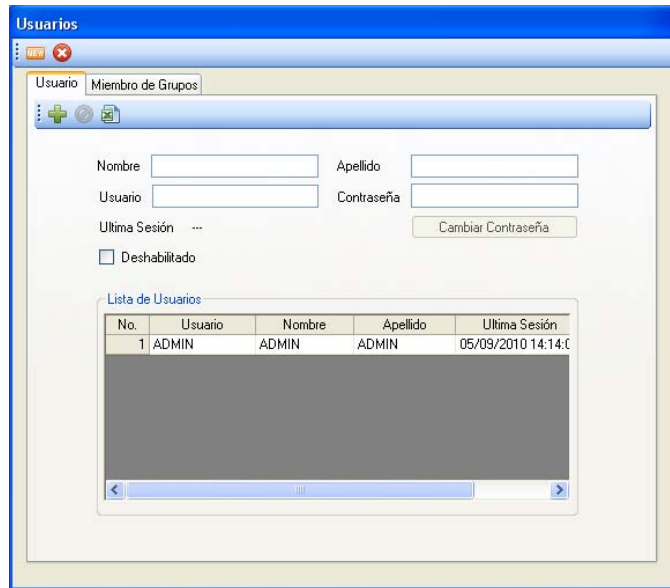


Gráfico 3.13. Usuarios

En la sección de Miembros de Grupo debemos seleccionar el usuario al que se le asignará al grupo. Una vez seleccionado el grupo para el usuario debemos hacer click en el botón Aceptar para poder grabar los cambios realizados.

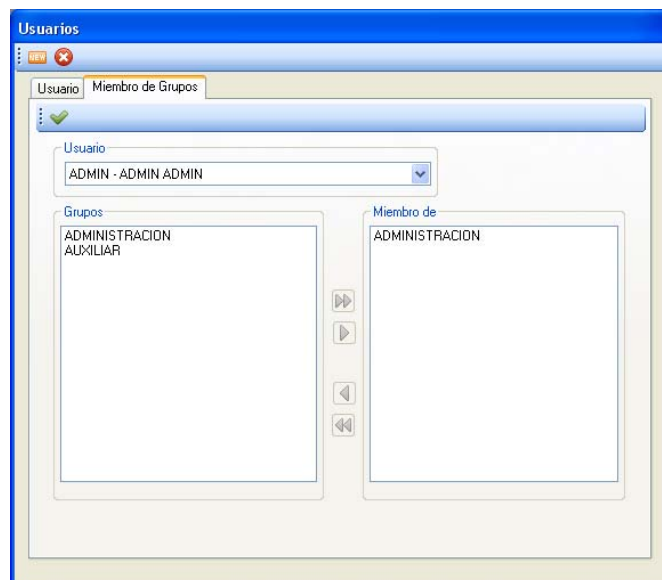
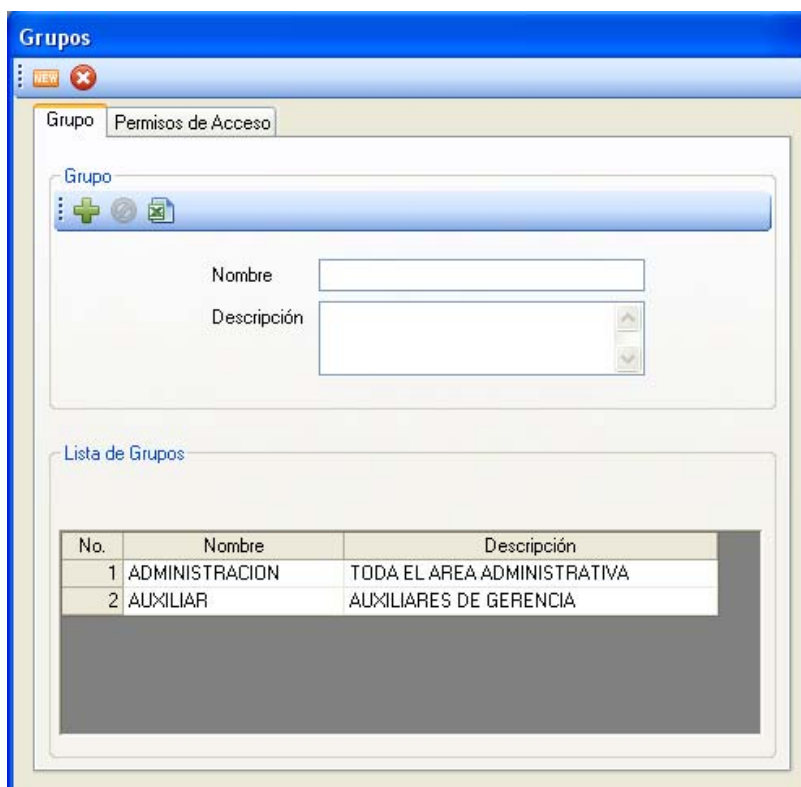


Gráfico 3.14. Miembros de los Grupos de Usuarios

3.10.2.2. Grupos

El submenú de grupos posee, al igual que los usuarios, 2 secciones que debemos configurar. La sección del grupo y la sección de permisos de acceso.

En la sección de Grupos es donde se crearán los grupos a los que se les asignarán los usuarios. Luego de la creación de cada grupo de trabajo es necesario hacer click en el botón Agregar para grabar los grupos creados.



The screenshot shows a software window titled "Grupos" with two tabs: "Grupo" and "Permisos de Acceso". The "Grupo" tab is active and contains a form with two input fields: "Nombre" and "Descripción". Below the form is a section titled "Lista de Grupos" containing a table with the following data:

No.	Nombre	Descripción
1	ADMINISTRACION	TODA EL AREA ADMINISTRATIVA
2	AUXILIAR	AUXILIARES DE GERENCIA

Gráfico 3.15. Grupos

En la sección de Permisos de Acceso se le asignarán los privilegios para cada grupo de trabajo. Primero se selecciona el grupo de trabajo al que se le habilitarán o deshabilitarán las opciones del manejo del sistema. En el recuadro debemos hacer un check, habilitando la opción del menú que deseamos el grupo tenga acceso. Luego damos click en el botón Aceptar para grabar los cambios realizados para cada grupo.

Nota: Todos los usuarios que estén dentro del grupo tendrán los mismos permisos de acceso.

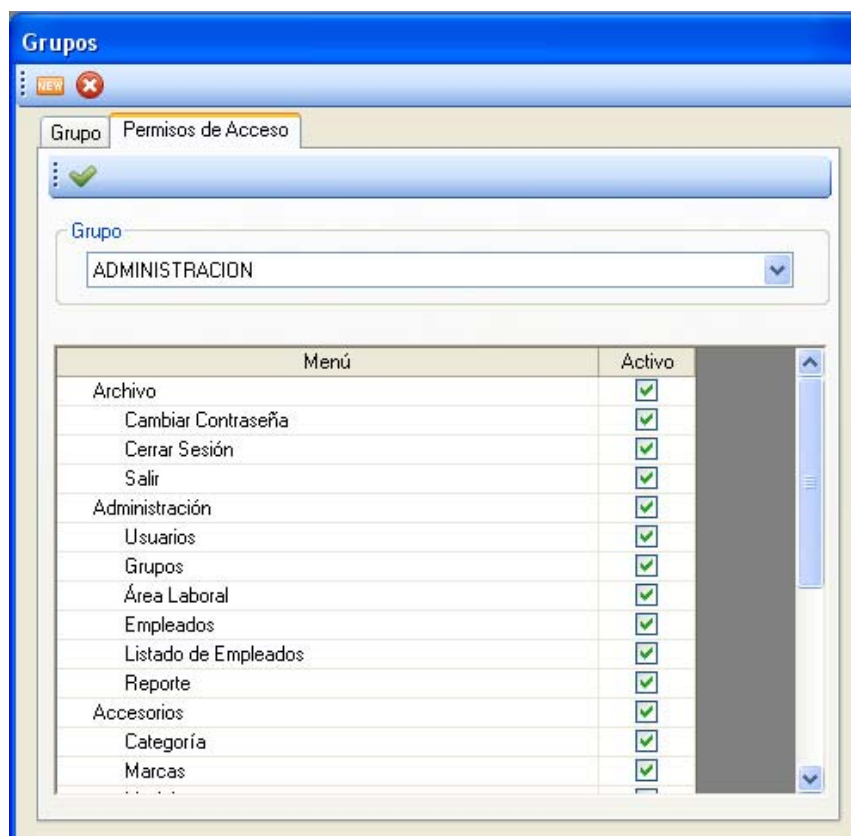


Gráfico 3.16. Permisos de Acceso para cada grupo

3.10.2.3. Áreas Laborales

El submenú de áreas laborales mostrará una pantalla donde se podrá crear todas las áreas que posee la empresa y a las cuales se les asignarán a los empleados.



The screenshot shows a software window titled "Áreas Laborales". The window contains two main sections. The top section, "Datos de Área Laboral", has a text input field for "Nombre" and a larger text area for "Descripción". The bottom section, "Lista de Áreas Laborales", contains a table with the following data:

No.	Nombre	Descripción
1	ADMINISTRACION	AREA ENCARGADA DE LA PART
2	SISTEMAS	AREA ENCARGADA DE TODOS L

Gráfico 3.17. Datos de Áreas Laborales

3.10.2.4. Empleados

El submenú de empleados abrirá la pantalla con la cual se podrá ingresar y controlar a los empleados.

Al ingresar un empleado se podrá ver que aparecerá en el listado con todos los empleados de la empresa. Al hacer doble click sobre un empleado, éste cargará sus datos, permitiendo al usuario modificar cualquier cambio necesario.

Empleados

Datos de Empleados

Nombre Apellido

Área Laboral CI/RUC

Dirección

Teléfono Celular

E-Mail

Lista de Empleados

No.	Nombre	Apellido	Área Laboral	CI/RUC	D
1	DIEGO	HIDALGO	ADMINISTRAC	1710547827	CALLE

Gráfico 3.18. Empleados

3.10.2.5. Listado de Empleados

Esta pantalla muestra solo un listado de todos los empleados registrados dentro del sistema. Permite también realizar búsquedas de empleados de acuerdo a su CI, Nombre o Área Laboral.

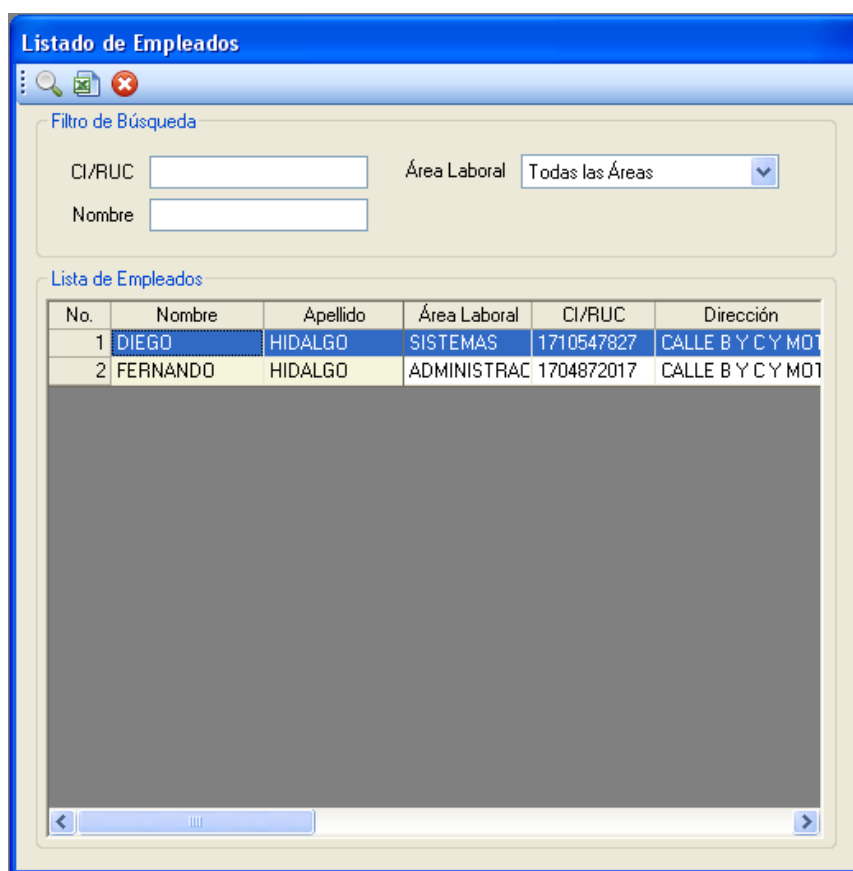
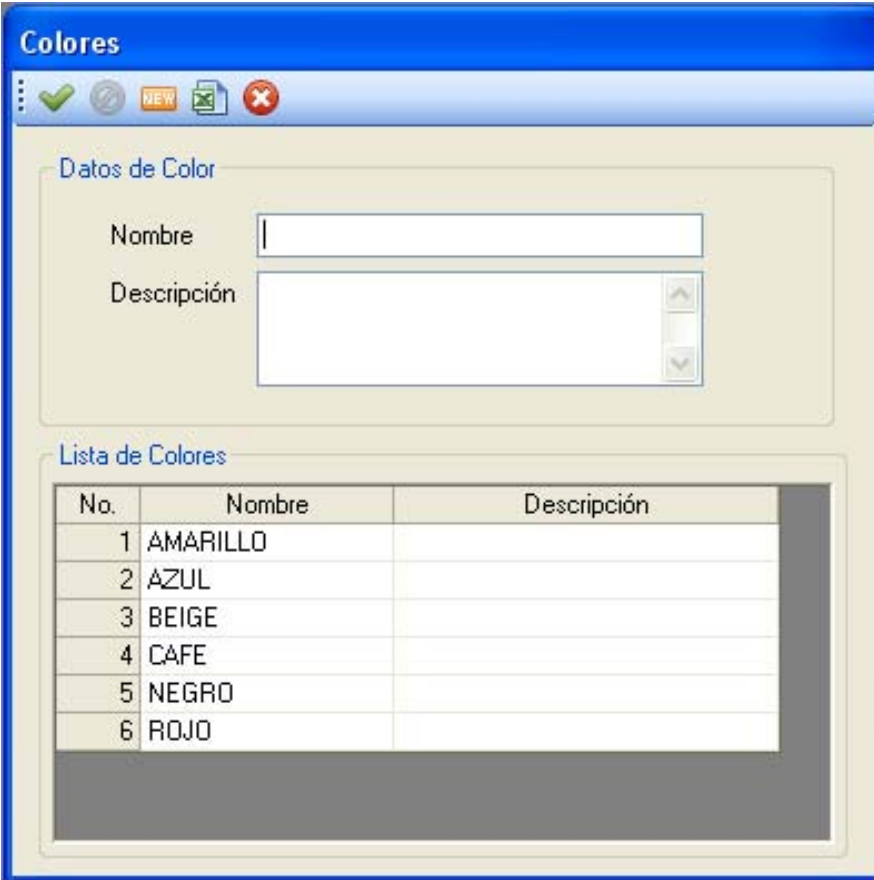


Gráfico 3.19. Listado de Empleados

3.10.3. Autos

3.10.3.1. Colores

Esta pantalla permite ingresar todos los colores que se puedan utilizar para el pintado de los vehículos.

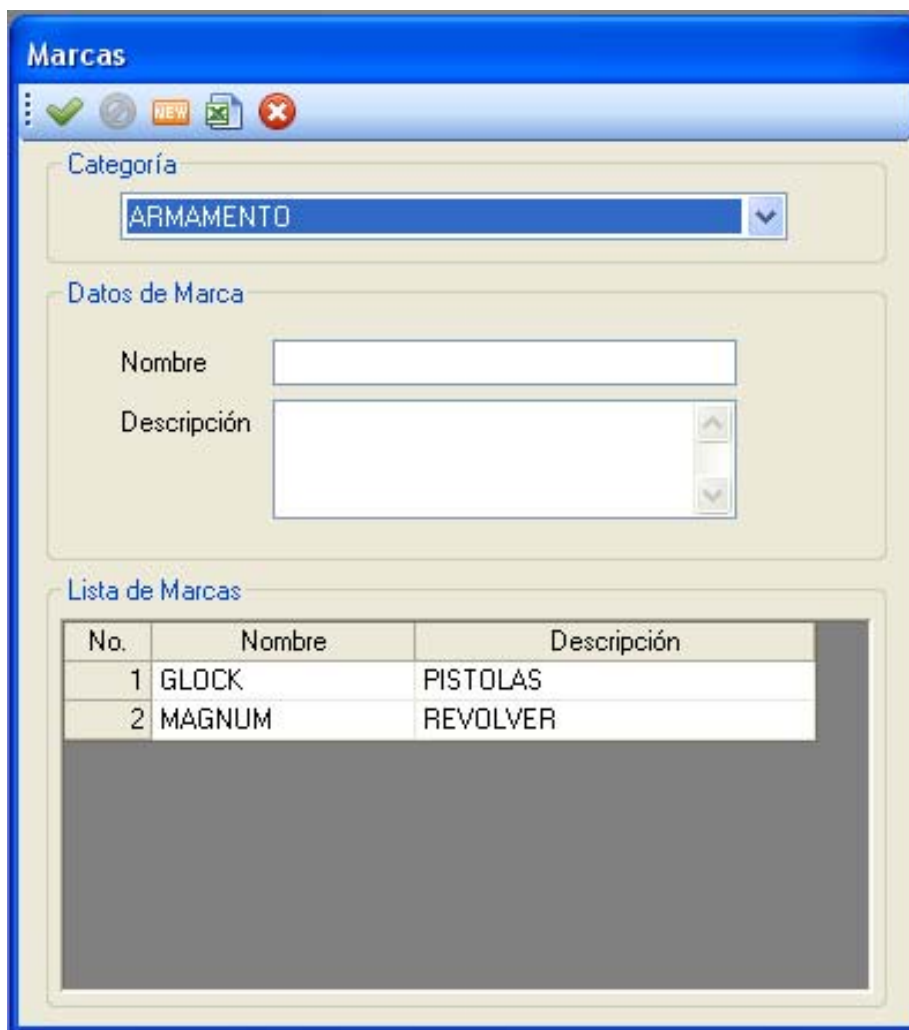


No.	Nombre	Descripción
1	AMARILLO	
2	AZUL	
3	BEIGE	
4	CAFE	
5	NEGRO	
6	ROJO	

Gráfico 3.20. Colores Autos

3.10.3.2. Marcas

Esta pantalla sirve para ingresar las posibles marcas de vehículos que se maneje dentro de la mecánica.



The screenshot shows a software window titled "Marcas" with a standard Windows-style title bar. Below the title bar is a toolbar with icons for save, refresh, new, print, and close. The main area is divided into three sections:

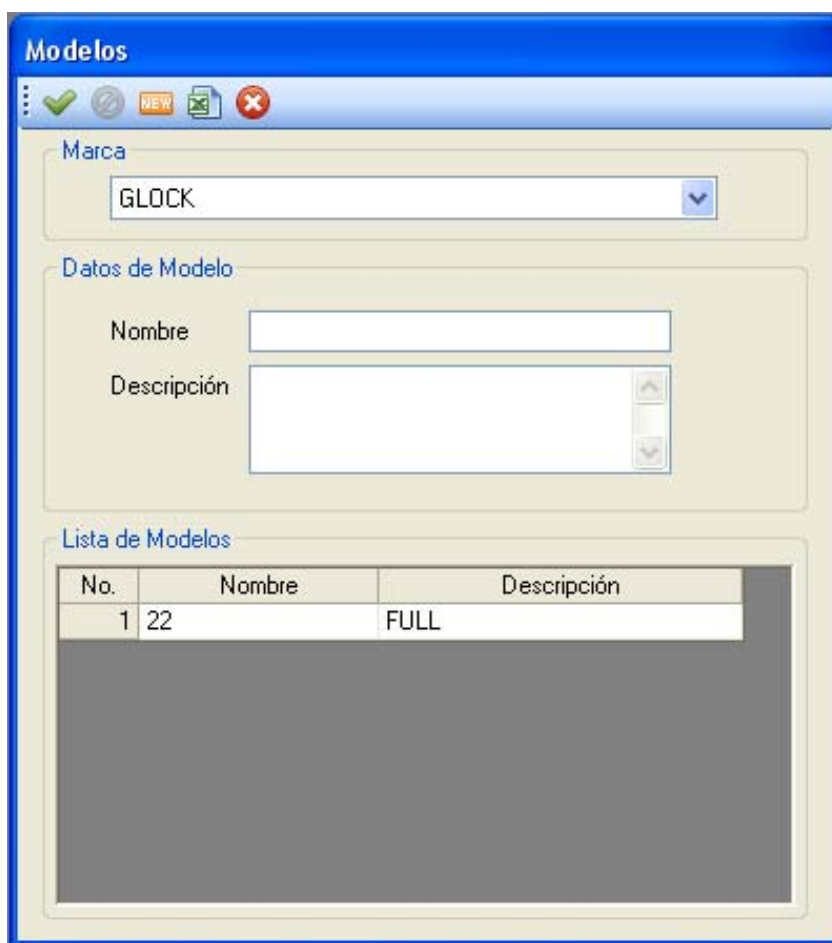
- Categoría:** A dropdown menu currently showing "ARMAMENTO".
- Datos de Marca:** A form with two fields: "Nombre" (a single-line text box) and "Descripción" (a multi-line text box with scrollbars).
- Lista de Marcas:** A table displaying a list of brands.

No.	Nombre	Descripción
1	GLOCK	PISTOLAS
2	MAGNUM	REVOLVER

Gráfico 3.21. Marcas Autos

3.10.3.3. Modelos

El submenú de modelos permite, al igual que las marcas, ingresar los modelos de los automóviles que sean registrados dentro de la mecánica.



The screenshot shows a software window titled "Modelos". At the top, there is a toolbar with icons for a checkmark, a refresh button, a "NEW" button, a save icon, and a close button. Below the toolbar, the window is divided into three sections:

- Marca:** A dropdown menu with "GLOCK" selected.
- Datos de Modelo:** A form with two fields: "Nombre" (empty) and "Descripción" (empty).
- Lista de Modelos:** A table with the following data:

No.	Nombre	Descripción
1	22	FULL

Gráfico 3.22. Modelos Autos

3.10.3.4. Listado De Autos

Esta pantalla muestra un listado completo de los vehículos ingresados. Muestra todos los vehículos que pertenecen, o se han asignado a cada cliente. La pantalla permite también realizar una búsqueda, de acuerdo al cliente, categoría, marca, modelo, color, año y color.

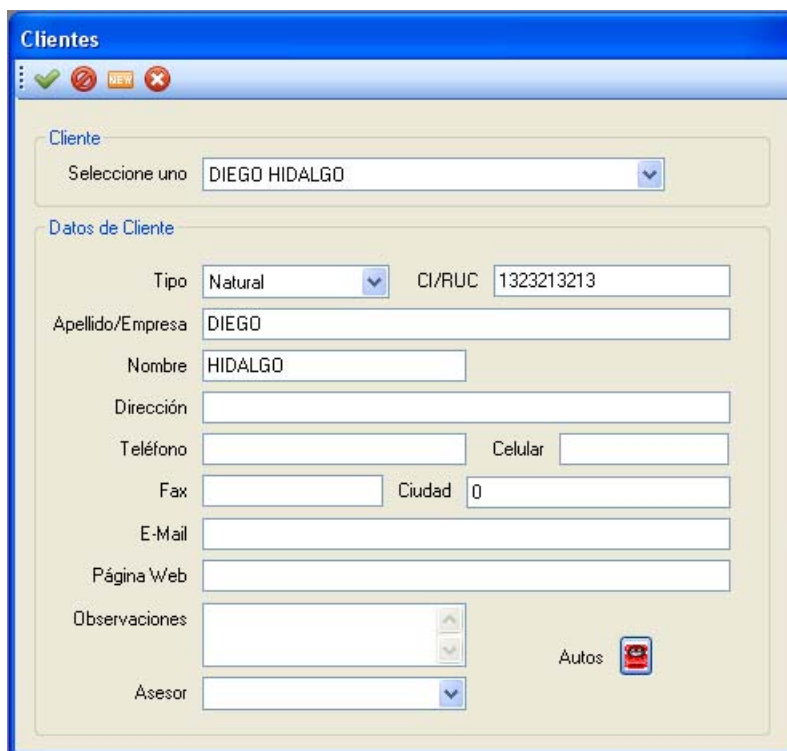
No.	Cliente	Marca	Modelo	Color	Año	Placa	Chasis
1	MORLA SANTIAGO	CHEVROLET	CORSA	AZUL	2008	PCD-96	SDA
2	MORLA SANTIAGO	CHEVROLET	CORSA	AZUL	2008	PCD-96	SDA
3	MORLA SANTIAGO	CHEVROLET	CORSA	AMARILLO	2009	151	
4	MORLA SANTIAGO	CHEVROLET	CORSA	AZUL	2008	PCD-96	SDA
5	MORLA SANTIAGO	FERRARI	DIABLO	AMARILLO	2009	111	
6	MORLA SANTIAGO	MAZDA	ALEGRO	AMARILLO	2010	PQR476	SDASD
7	SANTITO	CHEVROLET	CORSA	CAFE	2009	P000	4234
8	SDASD	CHEVROLET	SPARK	NEGRO	2009	21321	34324

Gráfico 3.23. Listado de Autos

3.10.4. Cientes/Proveedores

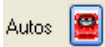
3.10.4.1. Clientes

Este submenú sirve para ingresar todos los clientes que la empresa pueda tener. Para revisar un cliente previamente ingresado es necesario seleccionarlo. Para poder ingresar un cliente es necesario ingresar la CI y Apellido/Empresa como mínimo.



The screenshot shows a software window titled "Clientes" with a standard Windows-style title bar. Below the title bar, there are several icons: a green checkmark, a red circle with a white 'X', a red circle with a white 'W', and a red circle with a white 'X'. The main content area is divided into two sections. The first section, labeled "Cliente", contains a dropdown menu with the text "Seleccione uno" and the value "DIEGO HIDALGO". The second section, labeled "Datos de Cliente", contains several input fields: "Tipo" (Natural), "CI/RUC" (1323213213), "Apellido/Empresa" (DIEGO), "Nombre" (HIDALGO), "Dirección", "Teléfono" and "Celular" (separate fields), "Fax" and "Ciudad" (separate fields), "E-Mail", "Página Web", "Observaciones" (a text area with scrollbars), and "Asesor" (a dropdown menu). At the bottom right of the form, there is a button labeled "Autos" with a small car icon.

Gráfico 3.24. Clientes

Una vez ingresado el cliente se puede registrar los vehículos que le pertenezcan. Oprimiendo  aparece una pantalla que permite realizar la

asignación de cada vehículo. Luego de seleccionar el cliente al que se le quiera asignar el vehículo, es necesario registrar los datos del auto, como lo son la placa, la marca el modelo, el chasis, etc.

The screenshot shows a software window titled "Autos" with a standard Windows-style title bar. The window is divided into three main sections:

- Ciente:** A dropdown menu showing "DIEGO HIDALGO".
- Datos de Auto:** A form with several input fields:
 - Marca: "Seleccione una..." (dropdown)
 - Modelo: "Seleccione uno..." (dropdown)
 - Año: empty text box
 - Placa: empty text box
 - Color: "Seleccione uno..." (dropdown)
 - Chasis: empty text box
 - Motor: empty text box
 - Observaciones: empty text area with scrollbars
- Lista de Autos:** A table with the following data:

No.	Marca	Modelo	Año	Placa	Color
1	MAZDA	ALEGRO	0		

Gráfico 3.25. Auto de cada cliente

3.10.4.2. Proveedores

Al igual que los clientes, este submenú sirve para poder ingresar los proveedores de la empresa. Funciona igual que la pantalla de clientes.

Proveedores

Proveedor
Seleccione uno: Nuevo Proveedor...

Datos de Proveedor

Tipo: Natural CI/RUC:

Apellido/Empresa:

Nombre:

Dirección:

Teléfono: Fax:

Ciudad:

E-Mail:

Página Web:

Observaciones:

Gráfico 3.26. Proveedores

3.10.4.3. Listado De Clientes/Listado De Proveedores

Esta pantalla sirve como listado tanto de los clientes como de los proveedores. En ambos listados es posible también realizar búsquedas, de acuerdo a la CI o al nombre.

The screenshot shows a software window titled "Listado de Clientes". At the top, there is a search filter section labeled "Filtro de Búsqueda" containing two text input fields: "CI/RUC" and "Nombre". Below this is a table titled "Lista de Clientes". The table has seven columns: "No.", "Tipo", "Apellido", "Nombre", "CI/RUC", "Dirección", and "Teléfono". The table body is currently empty, showing a grey background. A horizontal scrollbar is visible at the bottom of the table area.

Gráfico 3.27. Listado de Clientes

3.10.5. Inventario

3.10.5.1. Repuestos

Esta ventana permite registrar en el inventario todos los repuestos de la mecánica con su respectivo costo y cantidad. Al momento de registrar cada repuesto se ingresa un monto mínimo y máximo que debe existir dentro de la bodega, y al momento en que el repuesto llega al mínimo, el sistema indicará al usuario que se debe realizar un pedido de dicho repuesto. Tiene un listado de todos los repuestos ingresados a bodega.

Repuestos

Datos de Respuesto

Código: Nombre:

Grupo: Seleccione uno... Marca:

Costo: Max: Min:

Ubicación: Punto de Pedido

Descripción:

Lista de Repuestos

No.	Código	Nombre	Marca	Grupo	Costo	L
1	1234	ALGUNA COSA	TOSHIBA	MECÁNICA	12,00	FILA I

Gráfico 3.28. Repuestos

3.10.5.2. Servicios

Este submenú permite registrar todos los servicios que se realicen dentro de la mecánica. Se debe ingresar el número de mecánicos que se necesitan para cada tarea, así como también el tiempo estimado (en minutos) de lo que se demora en realizar dicho servicio. Al igual que en la parte de repuestos, esta ventana posee un listado de todos los servicios que provee la empresa.

The screenshot shows a software window titled "Servicios". The window contains the following elements:

- Datos de Servicio:** A form with several input fields:
 - Código:
 - Nombre:
 - Grupo: (dropdown menu)
 - Marca:
 - Costo:
 - # Mecánicos:
 - Tiempo Estimado: min
 - Descripción:
- Lista de Servicios:** A table with the following data:

No.	Código	Nombre	Marca	Grupo	Costo
1	4444	INTERNET	TV CABLE	JAJAJA	1213,40

Gráfico 3.29. Servicios

3.10.5.3. Grupos De Repuestos/Servicios

Esta pantalla permite crear todos los grupos y subgrupos tanto para los repuestos como para los servicios. Al crear un grupo principal, es necesario ingresar un código que permita diferenciar el grupo, y al momento de crear un subgrupo es necesario seleccionar el grupo al que pertenecerá.

Grupos de Repuestos/Servicios

Datos de Grupo

Grupo SubGrupo

Código: 02

Nombre:

Descripción:

Lista de Grupos

No.	Código	Nombre	Descripción
1	01	SERVICIOS	
2	01.01	PINTADO	

Gráfico 3.30. Grupos de Repuestos/Servicios

3.10.5.4. Lista de Precios

Muestra un listado de los servicios y repuestos con sus respectivos precios. Esta pantalla permite filtrar tanto los repuestos como los servicios, para que de esta manera se pueda obtener el precio de lo que se necesita.

No.	Producto	Descripción	Costo
1	CONEJO		12,00
2	PINTURA		35,00

Gráfico 3.31. Lista de Precios

3.10.5.5. Listado de Repuestos/Servicios

Muestra un listado de todos los repuestos y servicios de manera más detallada. Esta pantalla permite también realizar una búsqueda mediante el tipo (repuesto o servicio), código, nombre, grupo o marca. Para realizar la búsqueda es necesario primero seleccionar todos los filtros que se desea y luego oprimir el botón de buscar.

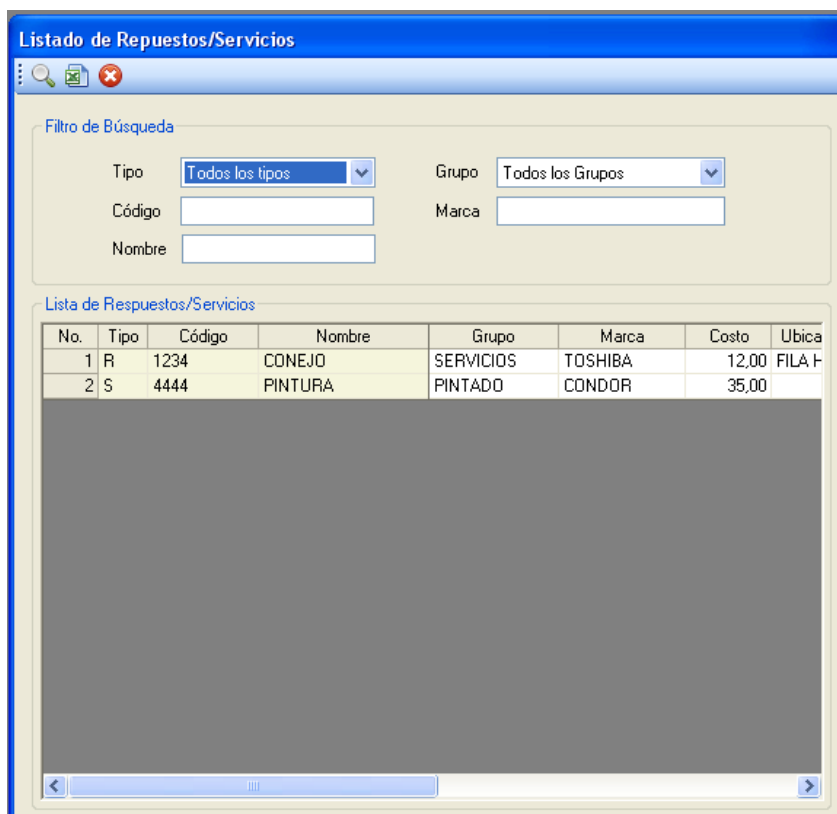
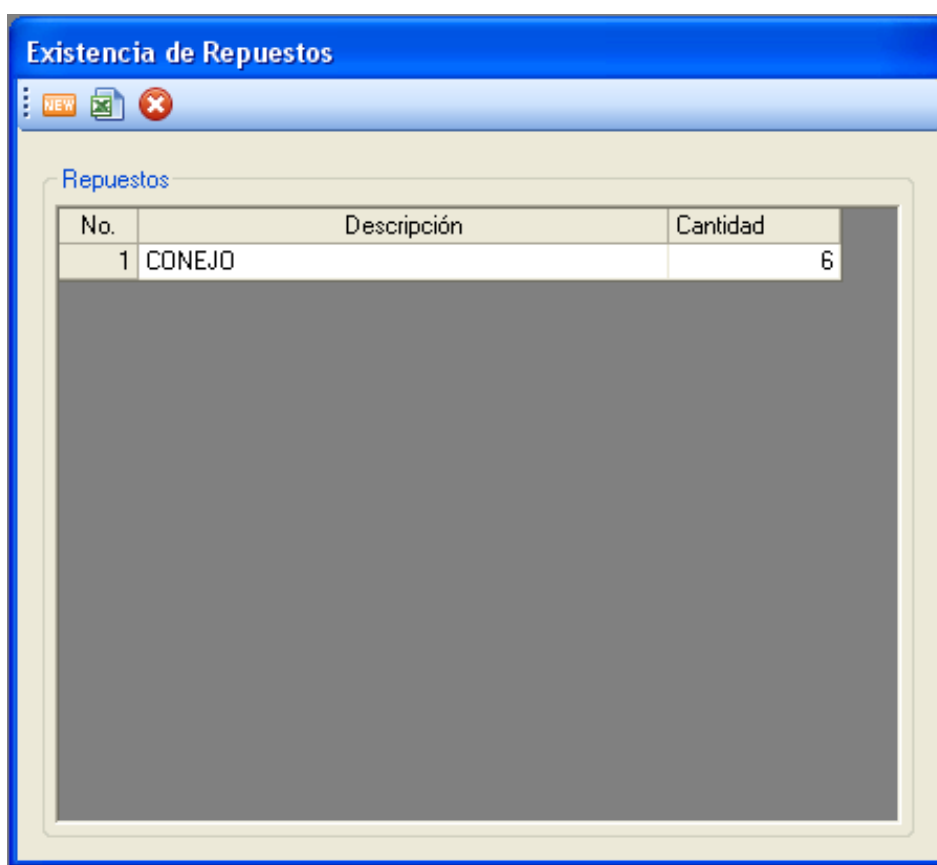


Gráfico 3.32. Listado de Repuestos/Servicios

3.10.5.6. Existencia de Inventario

Esta pantalla simplemente muestra los repuestos que tenemos en bodega. Esta pantalla permite realizar una actualización manual de los repuestos, haciendo doble click en el valor y permitiendo el cambio.



The screenshot shows a window titled "Existencia de Repuestos" with a standard Windows-style title bar. Below the title bar, there are three icons: a "NEW" button, a document icon, and a close button. The main content area is labeled "Repuestos" and contains a table with the following data:

No.	Descripción	Cantidad
1	CONEJO	6

Gráfico 3.33. Existencia de Repuestos en bodega

3.10.6. Ventas

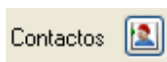
3.10.6.1. Aseguradoras

Esta pantalla permite el ingreso de las aseguradoras que tengan convenio con la mecánica. Al momento de ingresar la aseguradora, al realizar doble click en el listado sobre una aseguradora en general, cargará sus datos, permitiendo una posible modificación a cualquier dato erróneo.

No.	Nombre	CI/RUC	Dirección	Teléfono
1	SEGUROS COLONIAL	2132132721	Eloy Alfaro	3340964

Gráfico 3.34. Aseguradoras

Al cargar los datos de la aseguradora, se permite hacer oprimir el botón de contactos



, cargando una pantalla para poder ingresar los datos del contacto para dicha aseguradora, y en caso de no tener uno permite el ingreso de un nuevo contacto. Para el ingreso de nuevos registros es necesario seleccionar la aseguradora a la que pertenecerá el nuevo registro.

No.	Nombre	Apellido	CI/RUC	Teléfono	Celu
1	DIEGO	HIDALGO	1710547827	3340964	0999999

Gráfico 3.35. Contactos de cada Aseguradora

3.10.6.2. Asesores

Permite ingresar los datos de los asesores de la mecánica. Posee un listado de todos los registros previos, permitiendo seleccionarlos para una posible modificación.

The screenshot shows a software window titled "Asesores". The window has a blue title bar and a toolbar with icons for save, refresh, new, print, and close. Below the toolbar is a section titled "Datos de Asesores" containing a form with the following fields:

- Nombre:
- Apellido:
- CI/RUC:
- Código Auxiliar:
- Teléfono:
- Celular:
- E-Mail:

Below the form is a section titled "Lista de Asesores" containing a table with the following data:

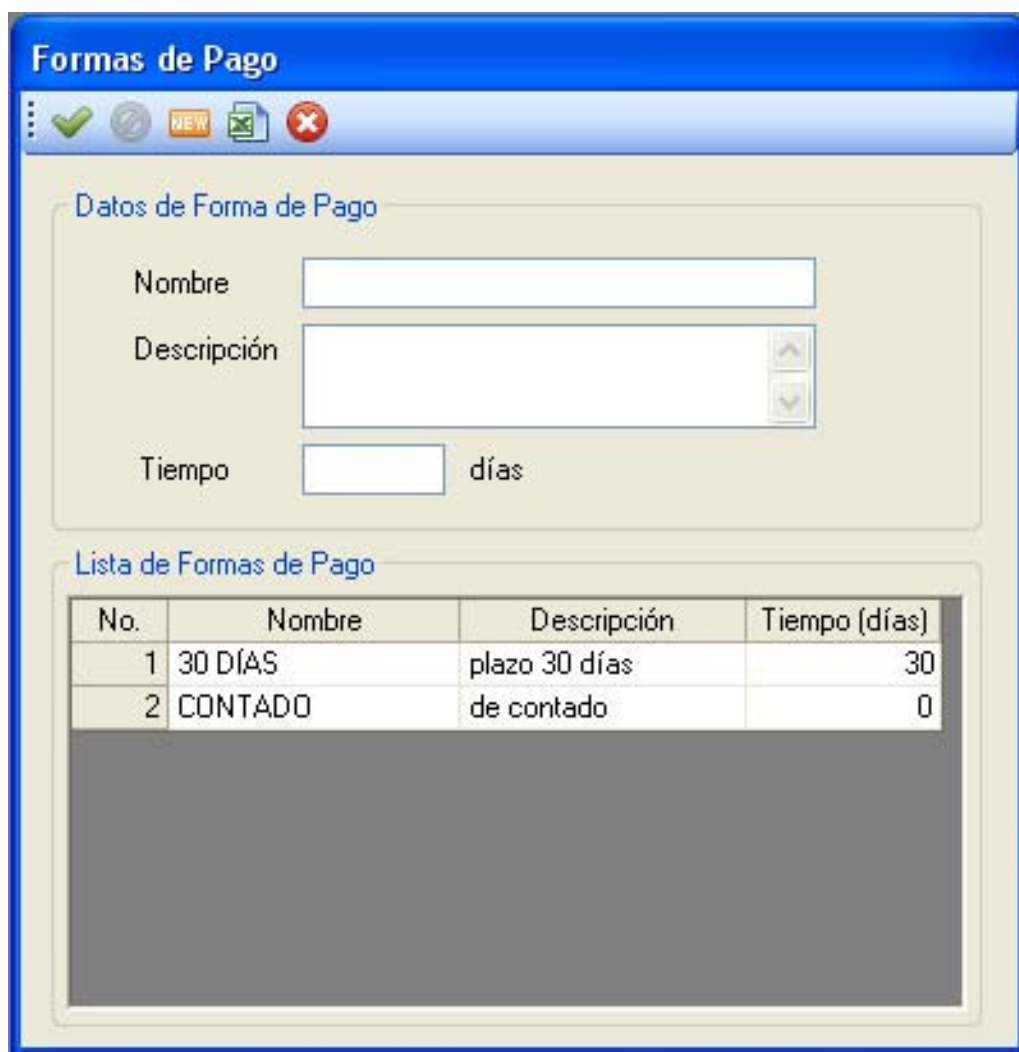
No.	Nombre	Apellido	CI/RUC	Teléfono	Celu
1	DIEGO	HIDALGO	1710547827		
2	SANTIAGO	MORLA	1710119577		

The table is displayed in a scrollable area with a horizontal scrollbar at the bottom.

Gráfico 3.36. Asesores

3.10.6.3. Formas de Pago

Simplemente permite el ingreso de las diferentes formas en las que el cliente puede realizar el pago a la mecánica.



The screenshot shows a software window titled "Formas de Pago" with a standard Windows-style title bar. Below the title bar, there are several icons: a green checkmark, a grey circular arrow, a red "NEW" button, a document icon, and a red "X" close button. The main content area is divided into two sections. The first section, "Datos de Forma de Pago", contains three input fields: "Nombre" (a text box), "Descripción" (a text box with scroll arrows on the right), and "Tiempo" (a text box followed by the word "días"). The second section, "Lista de Formas de Pago", contains a table with the following data:

No.	Nombre	Descripción	Tiempo (días)
1	30 DÍAS	plazo 30 días	30
2	CONTADO	de contado	0

Gráfico 3.37. Formas de Pago

3.10.6.4. Proforma

Esta pantalla permite al usuario ingresar los datos necesarios para generar una proforma del cliente. Esta pantalla nos permite seleccionar 2 opciones: Generar Proforma Manualmente y Modificar Proforma. Generar Proforma Manualmente permite al usuario crear una nueva proforma, ingresando tanto los datos completos del cliente como también el vehículo. Luego de ingresar los datos necesarios, se debe seleccionar los que se realizará en el vehículo, los servicios y los repuestos a utilizar.

Proforma carga los datos de una proforma previamente ingresada, permitiendo modificar sus datos erróneos.

The screenshot shows the 'Proforma' application window. At the top, there's a title bar with standard window controls. Below it, the 'Tipo de Proforma' section has a dropdown menu set to 'Generar Proforma Manualmente'. The 'Datos de la Proforma' section includes 'Fecha Ingreso' (2010-09-11), 'Aseguradora', and 'Destinatario' dropdowns, and a 'No. PROFORMA' field with the value '1'. The 'Datos del Cliente' section has fields for '* Cliente', 'Tipo' (Natural), 'Apellido', 'CI/RUC', '* Asesor', 'Asegurado', and 'Póliza'. The 'Datos del Auto' section includes 'Auto' (Nuevo Auto...), '* Marca' (Seleccione una...), '* Modelo' (Seleccione uno...), 'Año', 'Placas', 'Color', and 'Motor'. The 'Descripción de Trabajos' section features a table with columns: No., Descripción, Grupo, Precio Uni., Cantidad, Dcto. (\$), Dcto. (%), and Total. The table contains one row with '1' in the 'No.' column and a red 'X' icon in the 'Descripción' column. Below the table is an 'Observaciones' text area. On the right side, there's a summary section with 'Subtotal', 'IVA 12%', 'IVA 0%', and 'Total', all showing '0,00'.

No.	Descripción	Grupo	Precio Uni.	Cantidad	Dcto. (\$)	Dcto. (%)	Total
1							

Subtotal: 0,00
IVA 12%: 0,00
IVA 0%: 0,00
Total: 0,00

Gráfico 3.38. Proformas

3.10.6.5. Orden de Trabajo

Al igual que la pantalla de Proforma, esta pantalla permite ingresar los datos para generar cualquier orden de trabajo. Esta pantalla permite al usuario seleccionar 3 opciones de órdenes de trabajo: Generar Orden de Trabajo Manualmente, Generar Orden de Trabajo desde Proforma y Modificar Orden de Trabajo. Generar Orden de Trabajo Manualmente permite al usuario crear una NUEVA orden de trabajo, ingresando todos los datos del cliente, vehículo y los repuestos a usar y/o servicios que se realizarán. Generar Orden de Trabajo desde Proforma permite cargar los datos de una proforma ingresada para crear, a partir de estos datos, una nueva orden de trabajo y Modificar Orden de Trabajo permite cargar los datos de cualquier orden para realizar cualquier modificación necesaria.

Orden de Trabajo

Tipo de Proforma
Generar Orden de Trabajo Manualmente

Datos de la Orden de Trabajo
Fecha Ingreso: 2010-09-11
No. ORDEN: 2

Datos del Cliente
Tipo: Natural
Entrega del Auto
Auto: Nuevo Auto...
Marca: Seleccione una...
Modelo: Seleccione uno...
Año: Placas: Chasis: Motor:

Descripción de Trabajos

No.	Descripción	Grupo	Precio Uni.	Cantidad	Docto. (\$)	Docto. (%)	Total
1							

Observaciones:

Subtotal: 0.00
IVA 12%: 0.00
IVA 0%: 0.00
Total: 0.00

Gráfico 3.39. Orden de Trabajo

3.10.6.6. Generar Factura

Esta pantalla permite, a base de las órdenes de trabajo, generar las facturas sobre los trabajos realizados en el vehículo, indicando los repuestos utilizados y los servicios realizados. En la sección superior se erigirá la orden de trabajo y en la sección inferior se mostrará el detalle de la factura.

No. Orden	Fecha	Cliente	Auto	Observaciones
-----------	-------	---------	------	---------------

No.	Descripción	Precio Uni.	Cantidad	Dcto. (\$)	Dcto. (%)	Total	IVA
-----	-------------	-------------	----------	------------	-----------	-------	-----

Subtotal	0,00
IVA 12%	0,00
IVA 0%	0,00
Total	0,00

Gráfico 3.40. Facturar Orden de Trabajo

3.10.6.7. Reportes

Permite generar reportes basados tanto en las facturas ingresadas como en las órdenes de trabajo.

3.11. PASO A PASO USO DEL SOFTWARE

- 1) Buscar icono de acceso al software



Gráfico 3.41. Acceso

- 2) Doble click icono de acceso, despliega cuadro de dialogo, ingresar usuario y contraseña

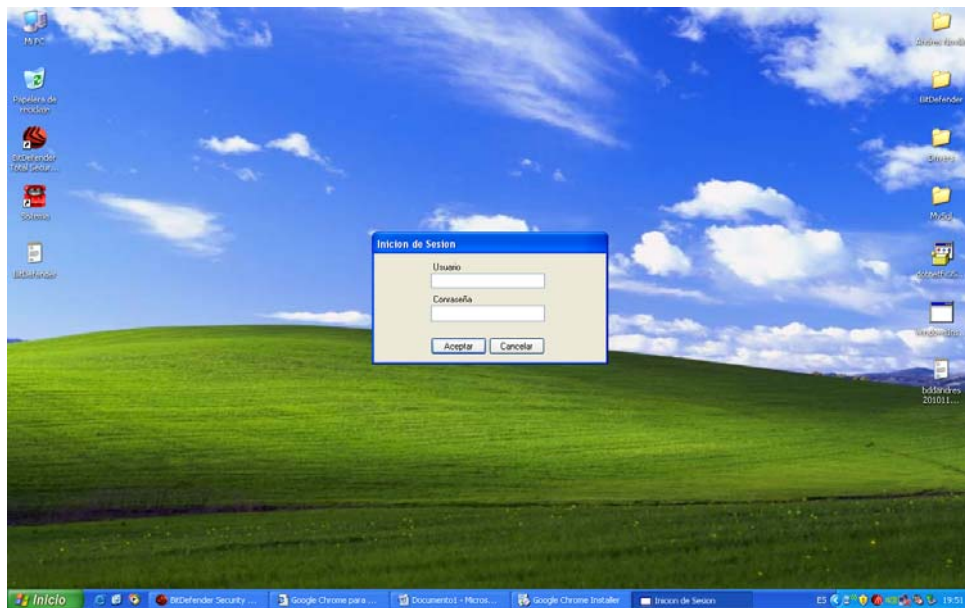


Gráfico 3.42. Inicio

3) Desplegar sub menú grupos en menú administración

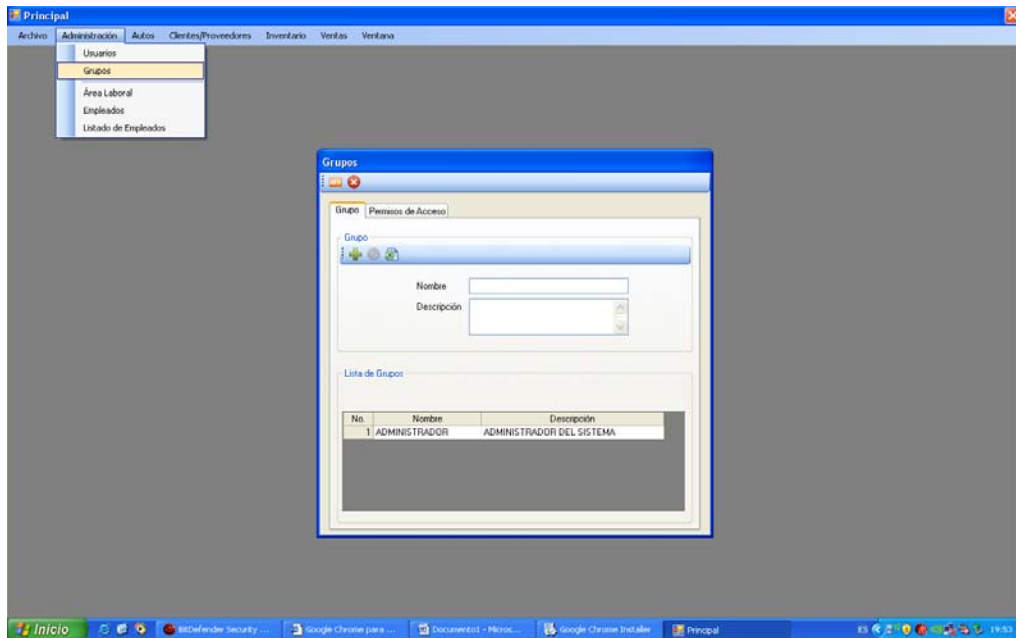


Gráfico 3.43. Grupos

4) Crear grupo para manejo de perfiles

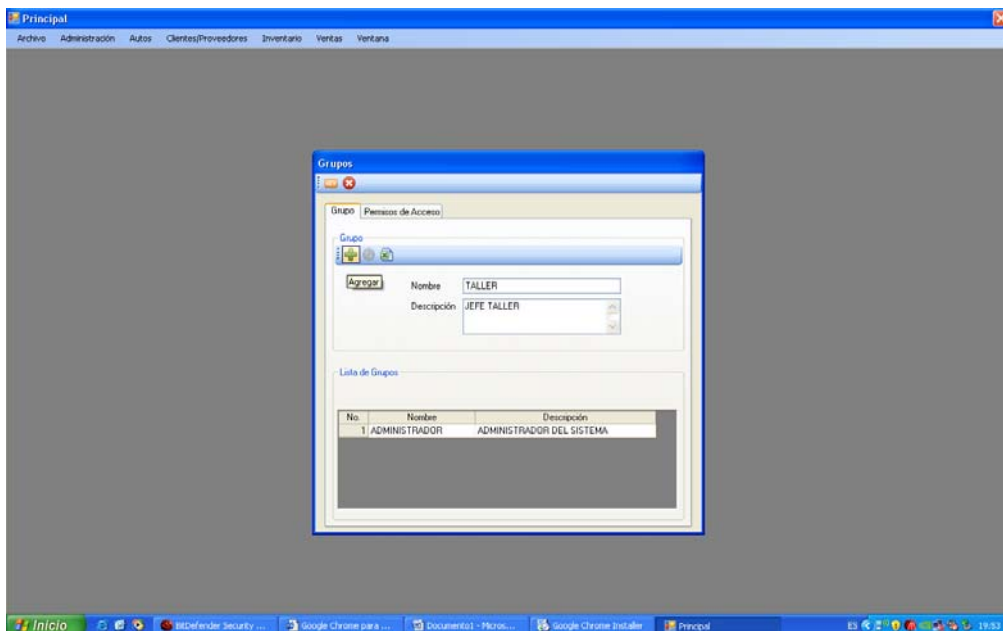


Gráfico 3.44. Crear grupo

5) Asignar perfiles de usuarios a grupos

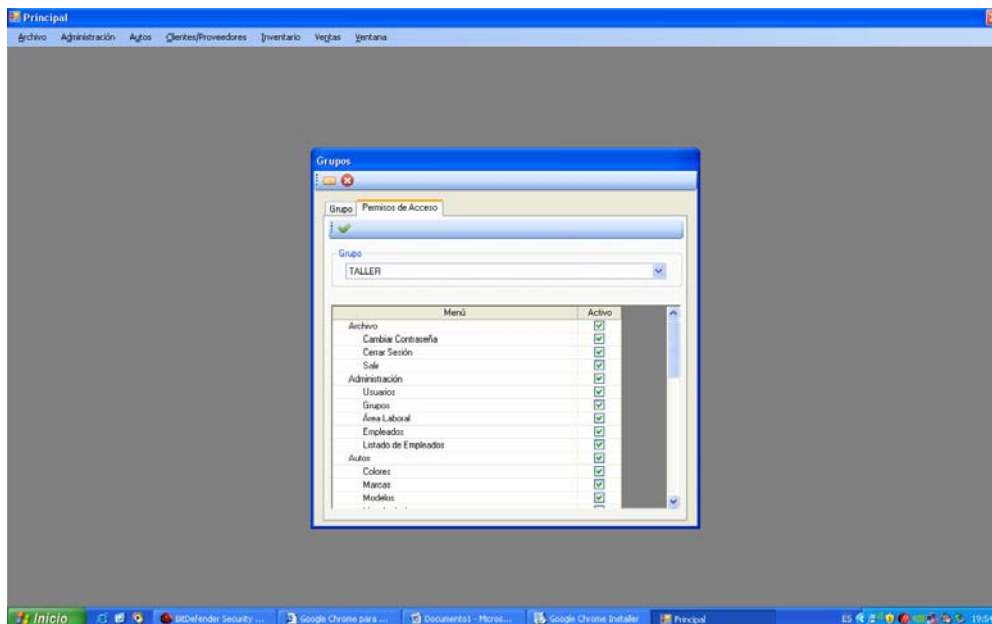


Gráfico 3.45. Perfiles grupo

6) Abrir menú usuarios

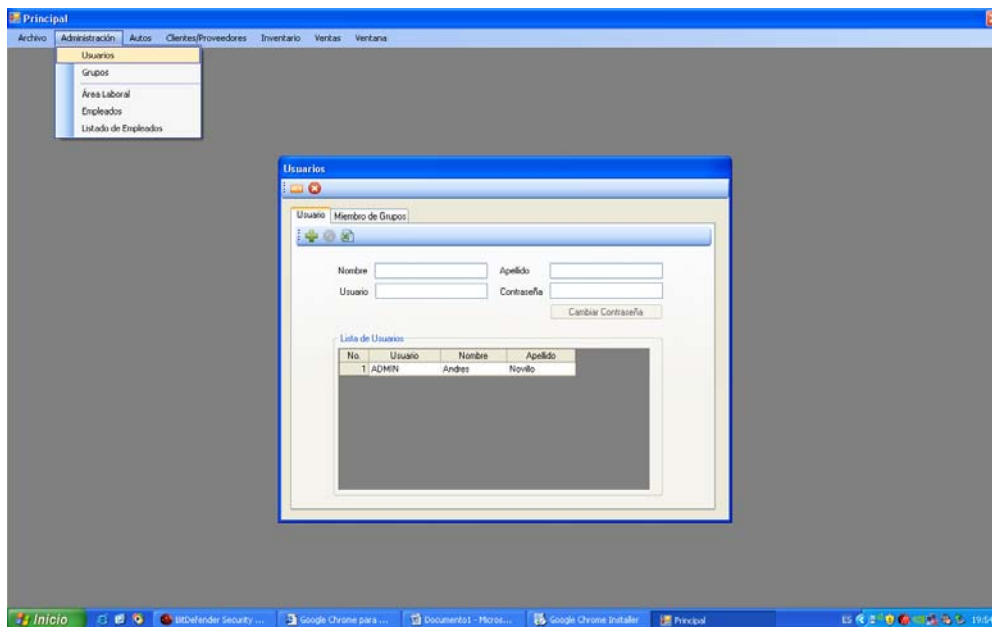


Gráfico 3.46. Usuarios

7) Agregar datos usuario

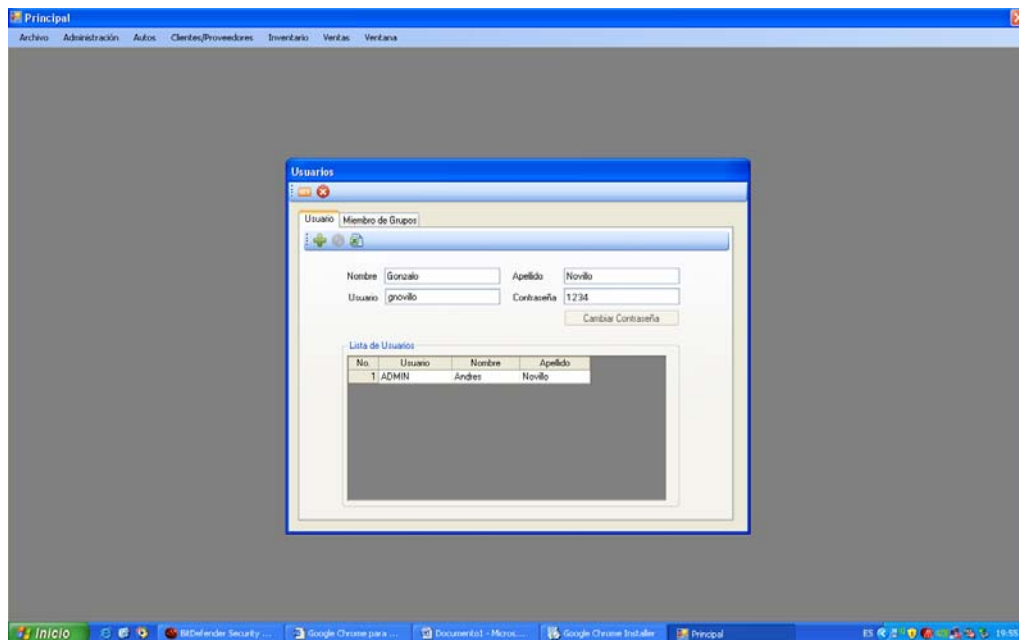


Grafico 3.47. Datos usuario

8) Agregar usuario a grupo

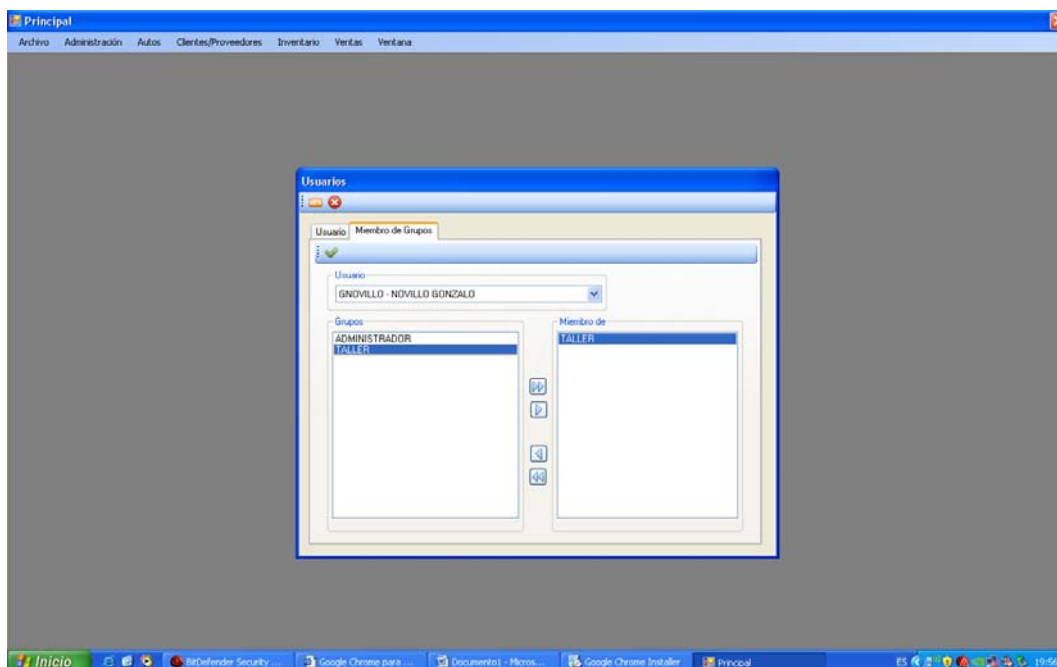


Gráfico 3.48. Asignación usuario a grupo

9) Abrir área laboral

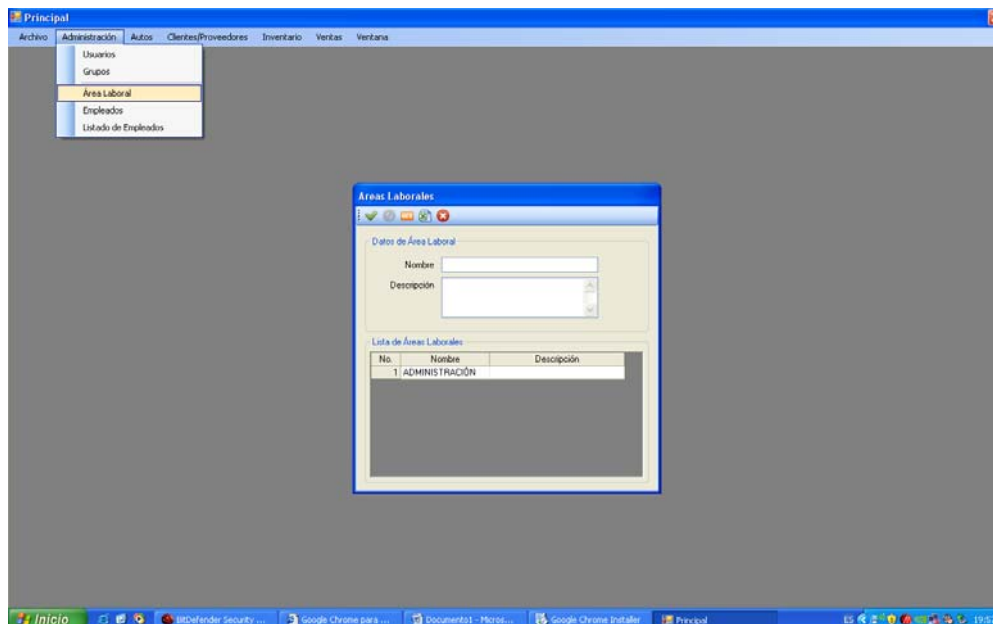


Gráfico 3.49. Área laboral

10) Creación Área Laboral

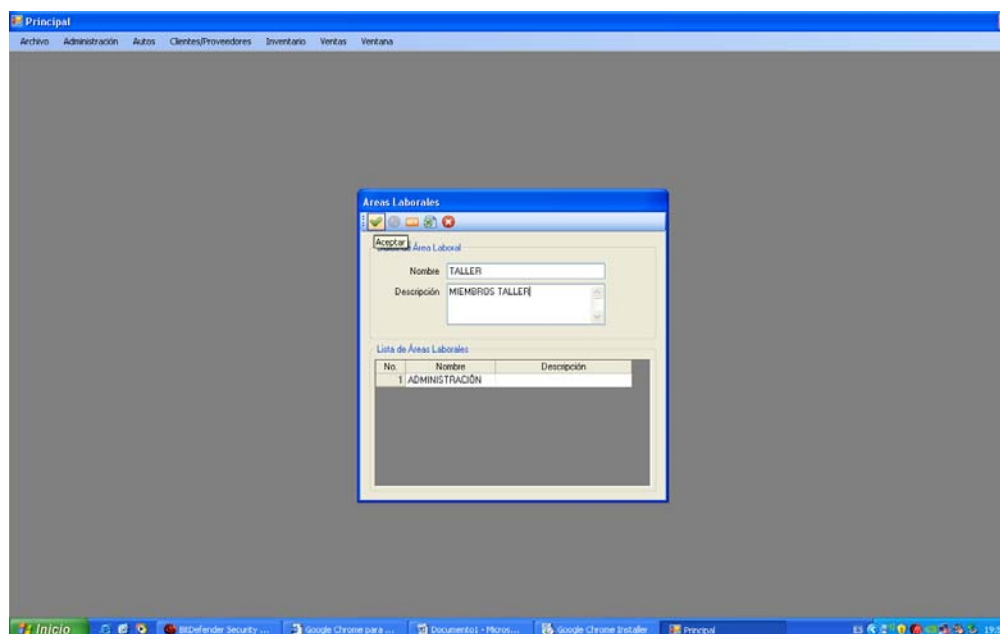


Gráfico 3.50 Creación Área Laboral

11) Abrir menú empleados

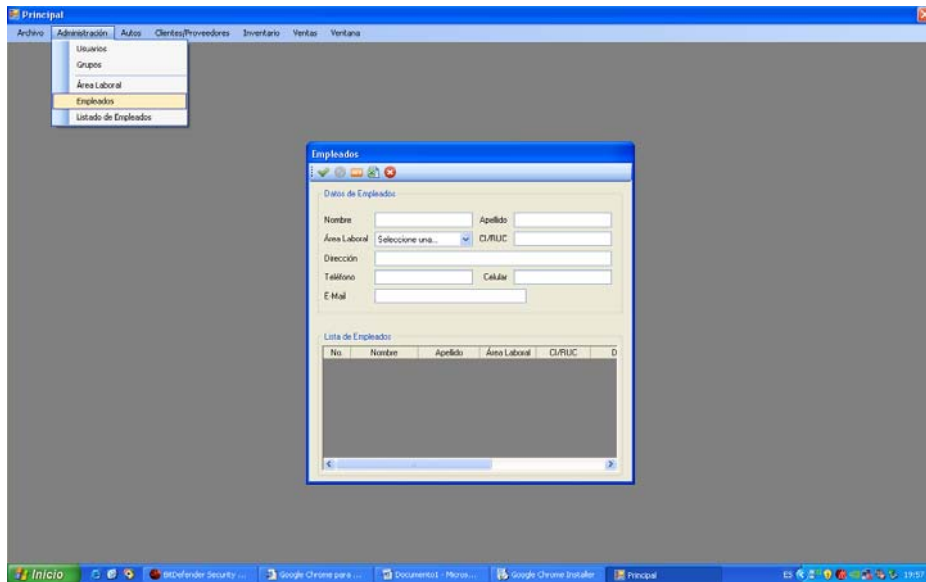


Gráfico 3.51. Empleados

12) Grabar nuevo empleado

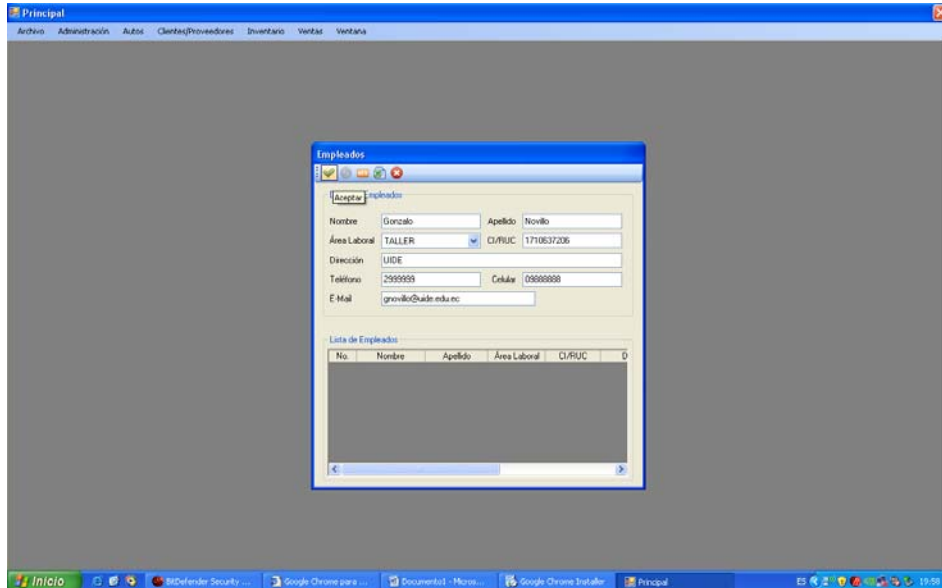


Gráfico 3.52. Grabar nuevo empleado

13) Visualizar lista de empleados

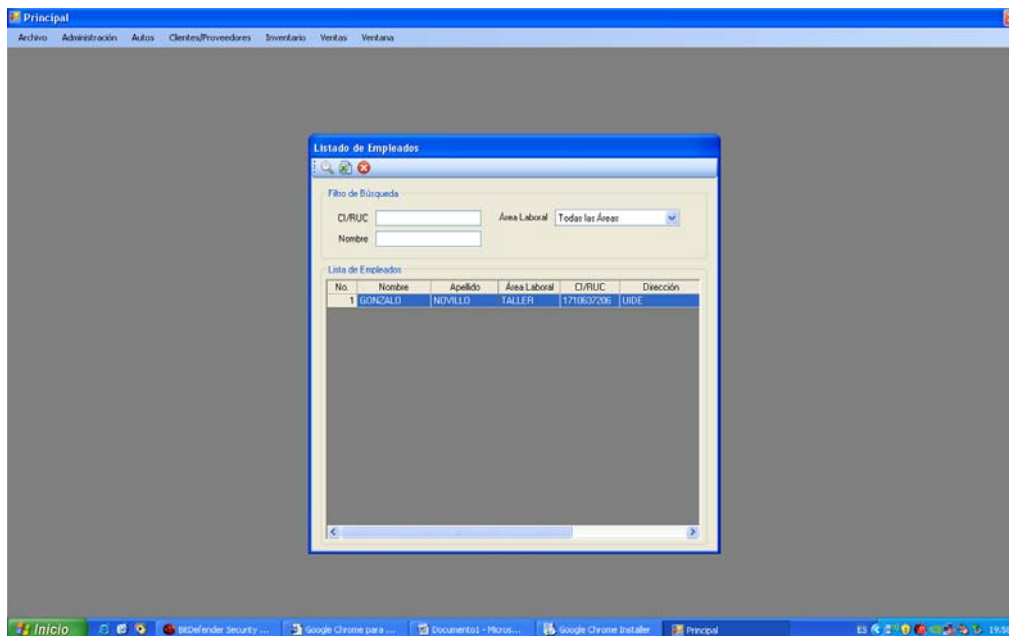


Gráfico 3.53. Lista de empleados

14) Abrir menú colores

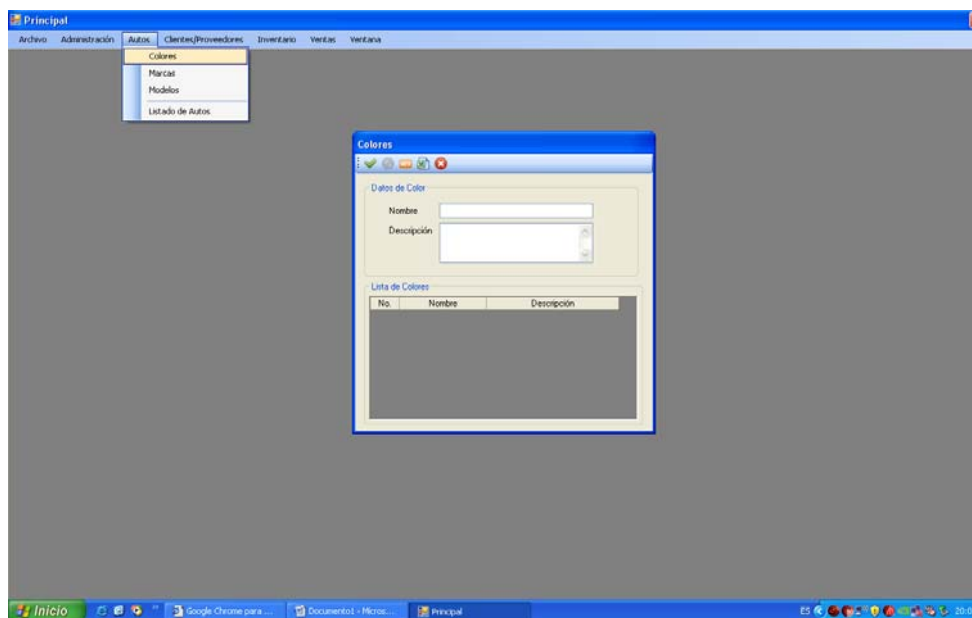


Gráfico 3.54. Colores

15) Abrir menú marcas

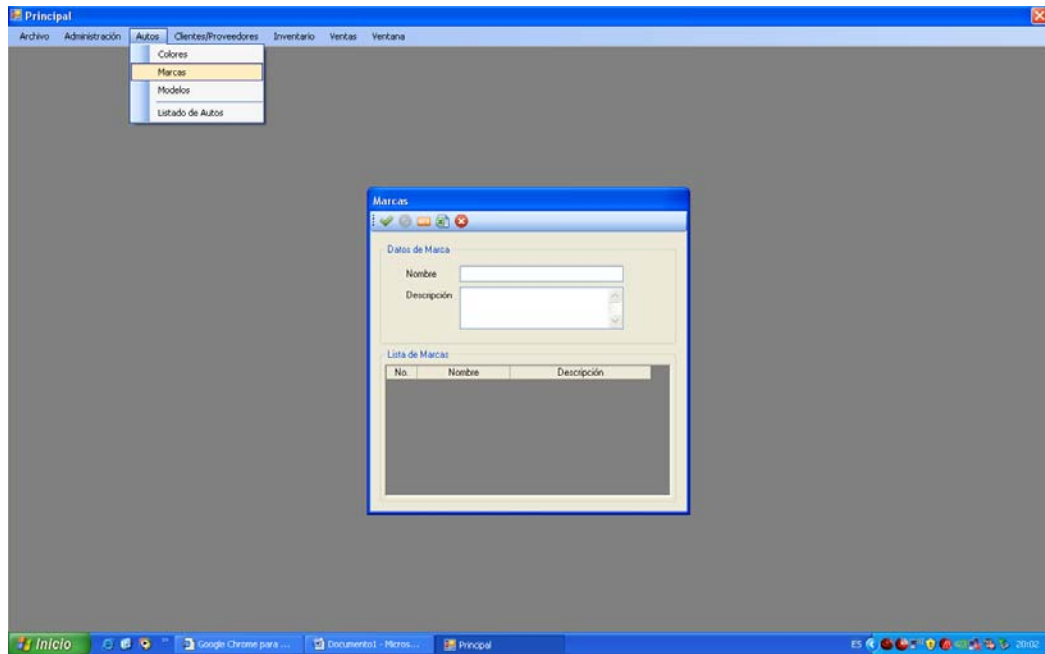


Gráfico 3.55. Marcas

16) Abrir menú modelos

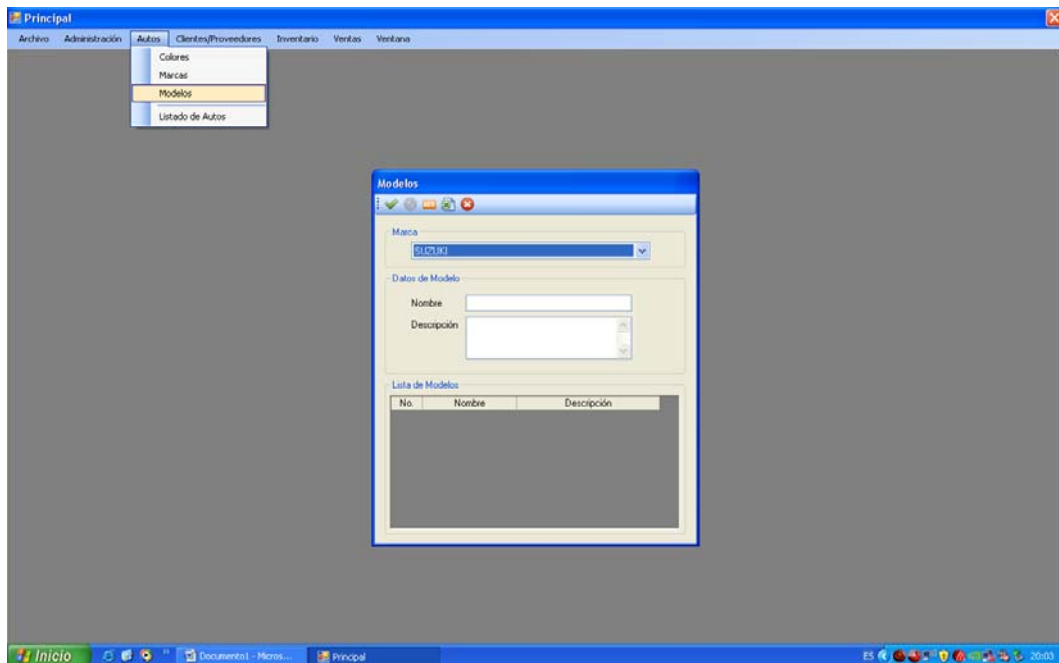


Gráfico 3.56. Modelos

17) Ingresar datos modelos

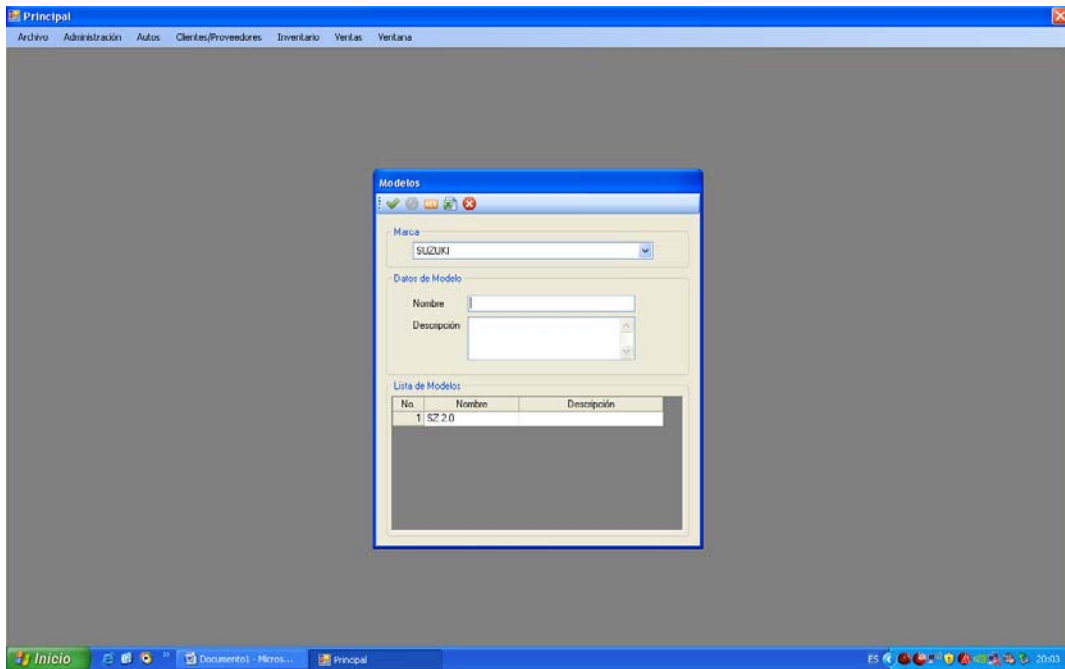


Gráfico 3.57. Ingresar datos modelos

18) Registrar información asesor

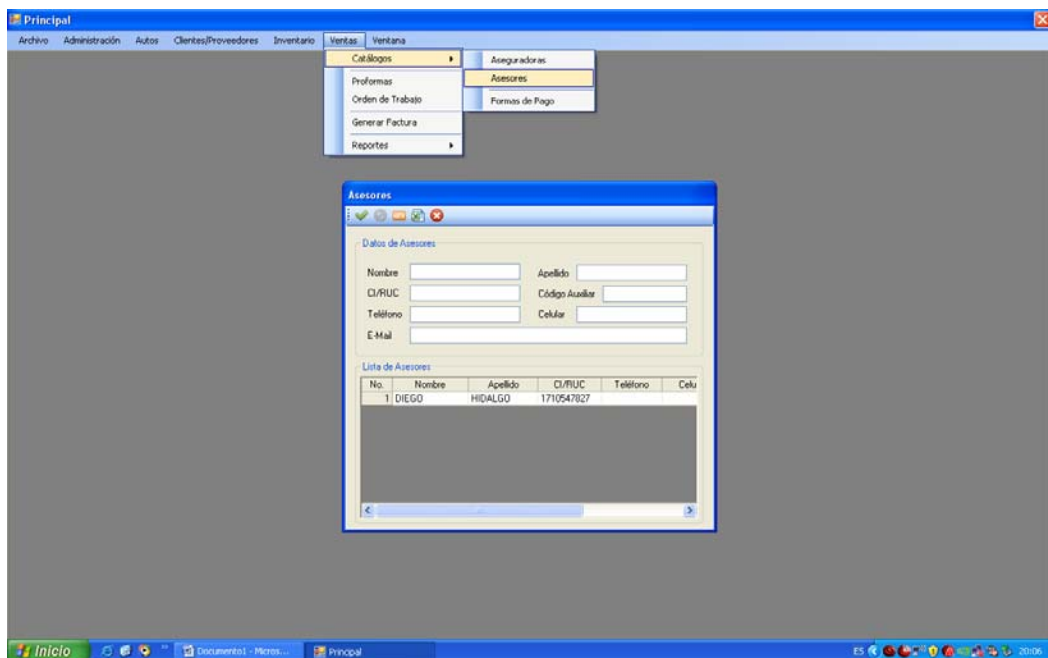


Gráfico 3.58. Asesor

19) Registrar información aseguradoras

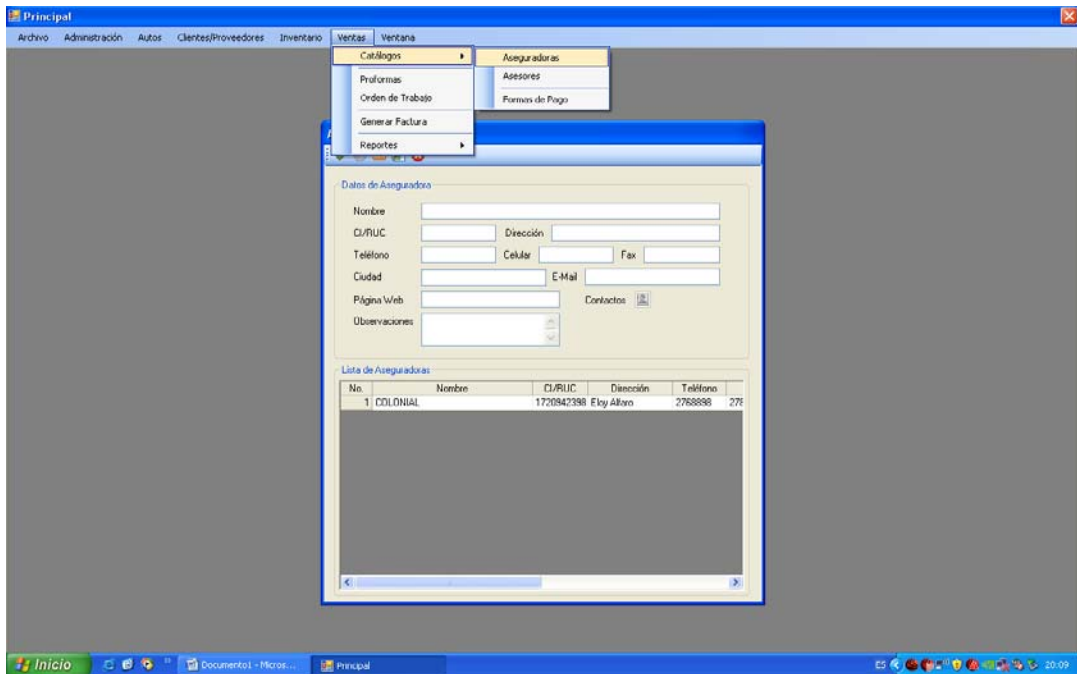


Gráfico 3.59. aseguradoras

20) Confirmación de registro aseguradoras

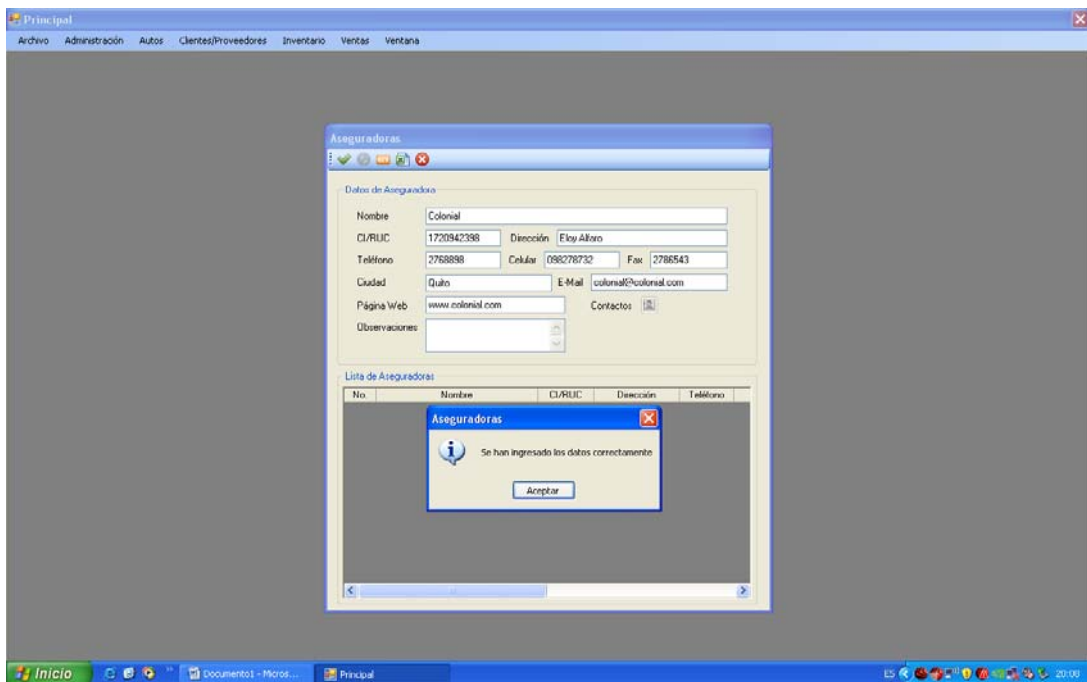


Gráfico 3.60. Confirmación aseguradora

21) Creación información cliente

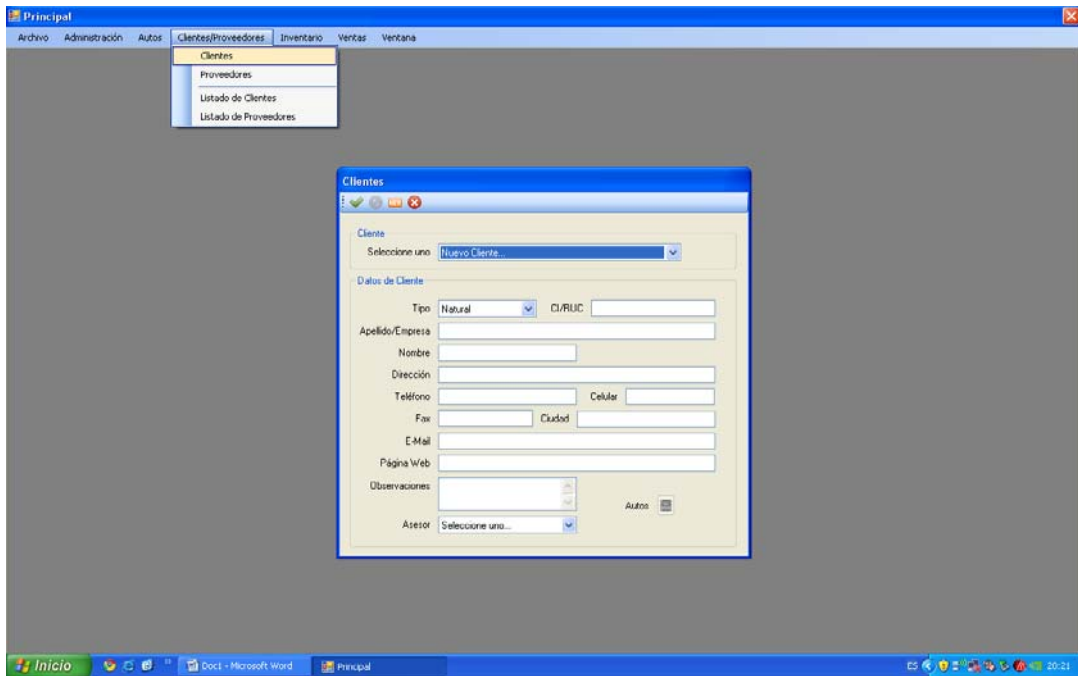


Gráfico 3.61. Creación cliente

22) Registro automóviles por cliente

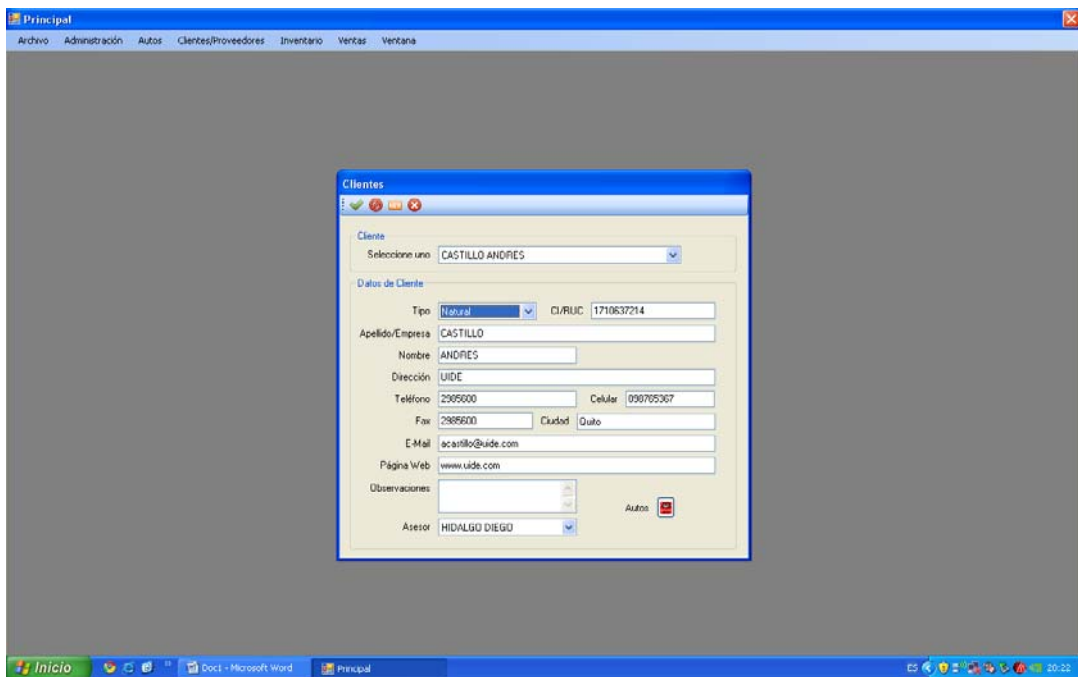


Gráfico 3.62. Automóviles por cliente

23) Ventana para registrar automóvil por cliente

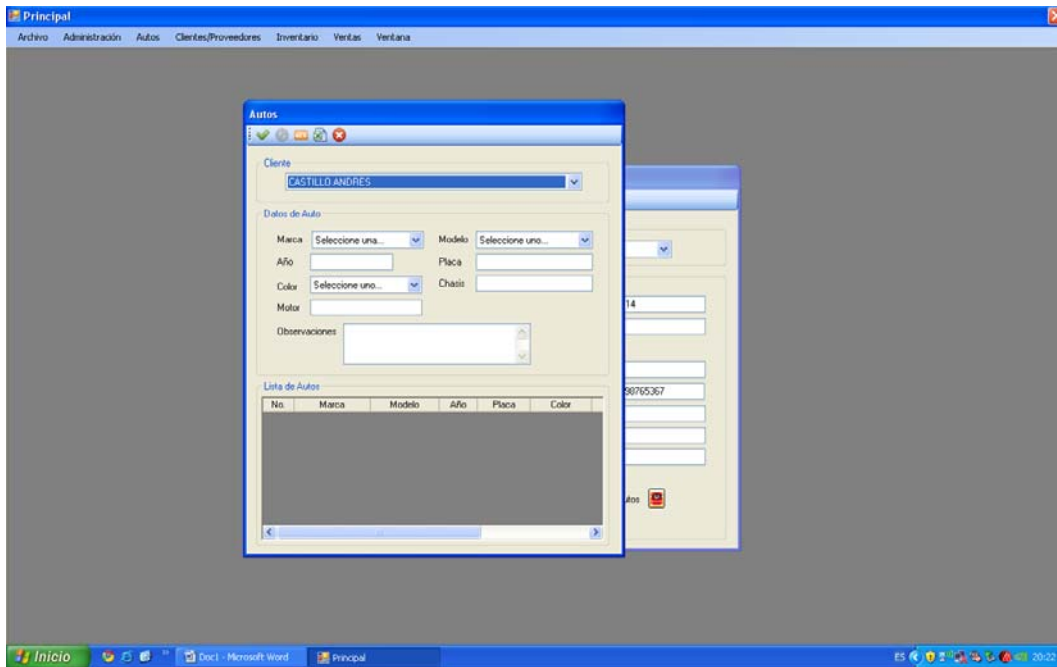


Gráfico 3.63. Automóvil por cliente

24) Confirmación vehículo por cliente

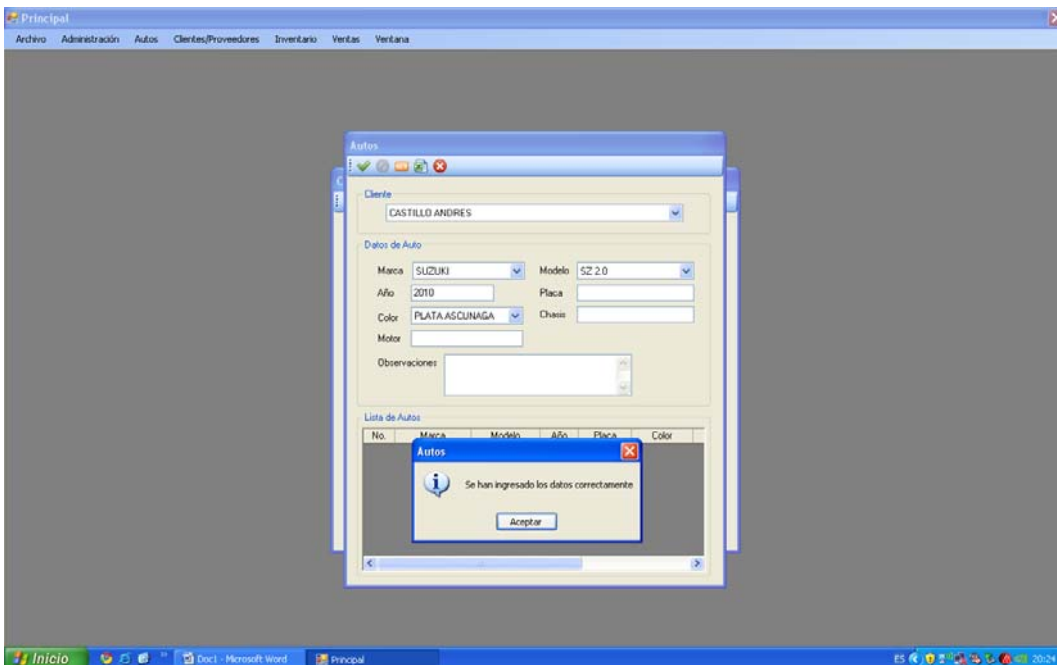


Gráfico 3.64. Confirmación vehículo por cliente

25) Registro información proveedores

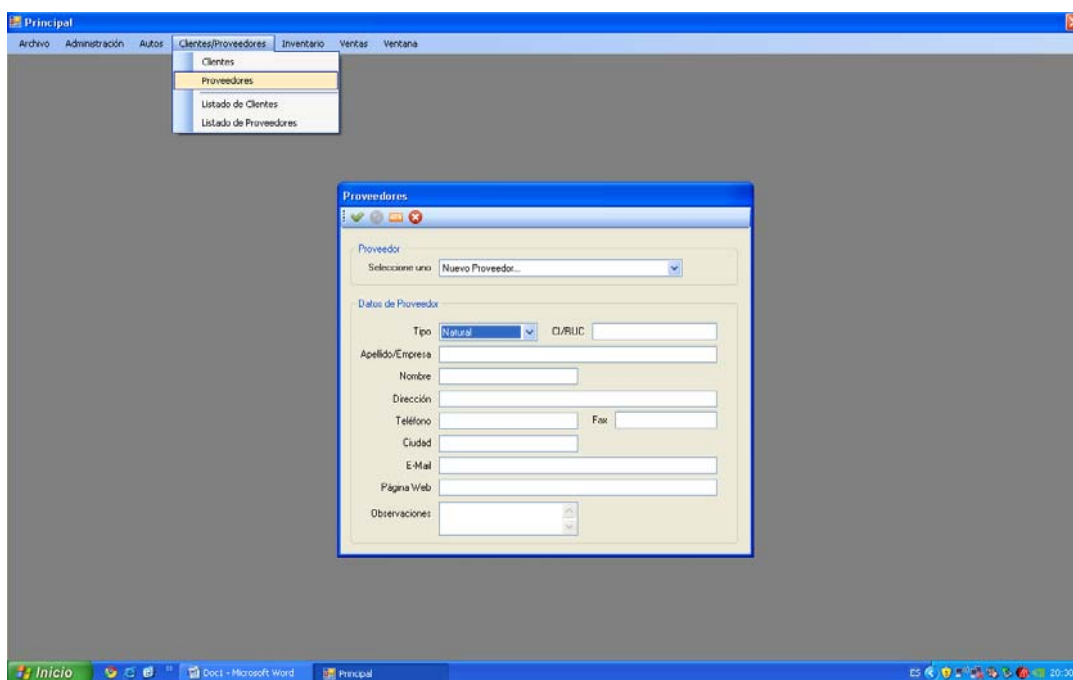


Gráfico 3.65. Proveedores

26) Confirmación registro proveedores

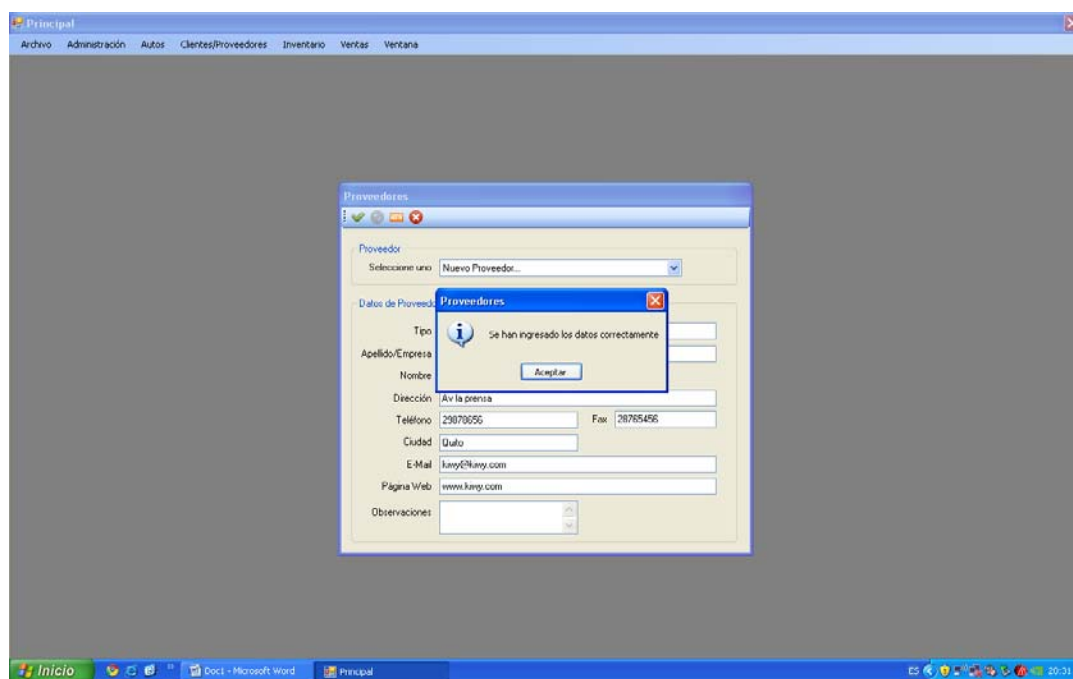


Gráfico 3.66. Registro roveedores

27) Creación grupos de servicios / materiales

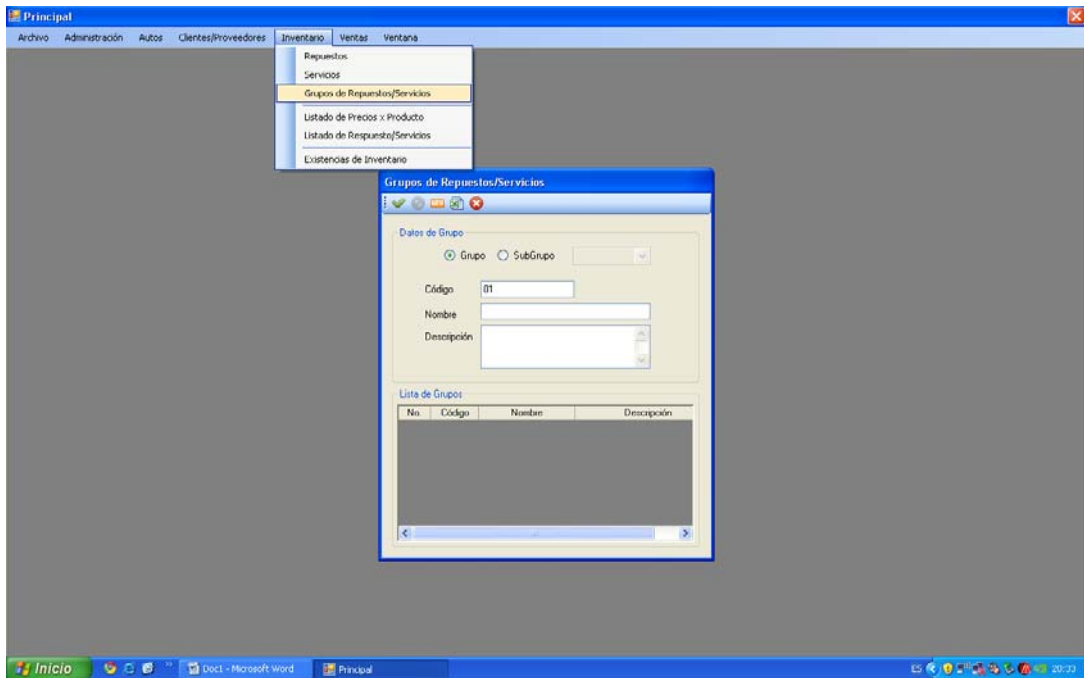


Gráfico 3.67. Servicios / Materiales

28) Confirmación creación grupo servicios

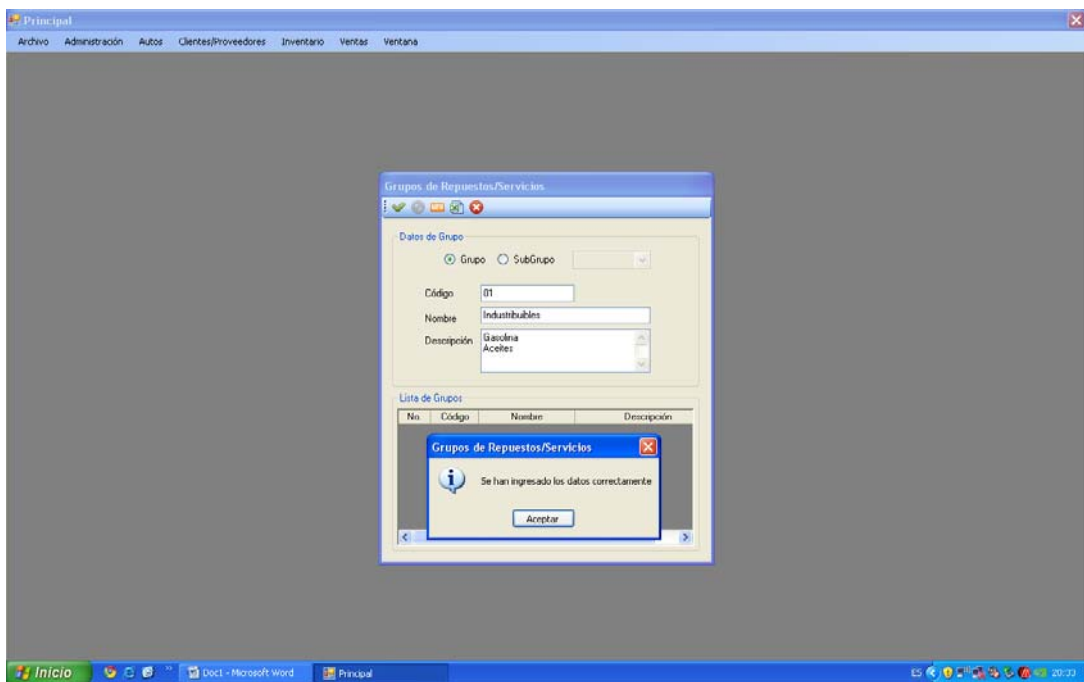


Gráfico 3.68. Confirmación creación grupo

29) Creación sub grupo servicios

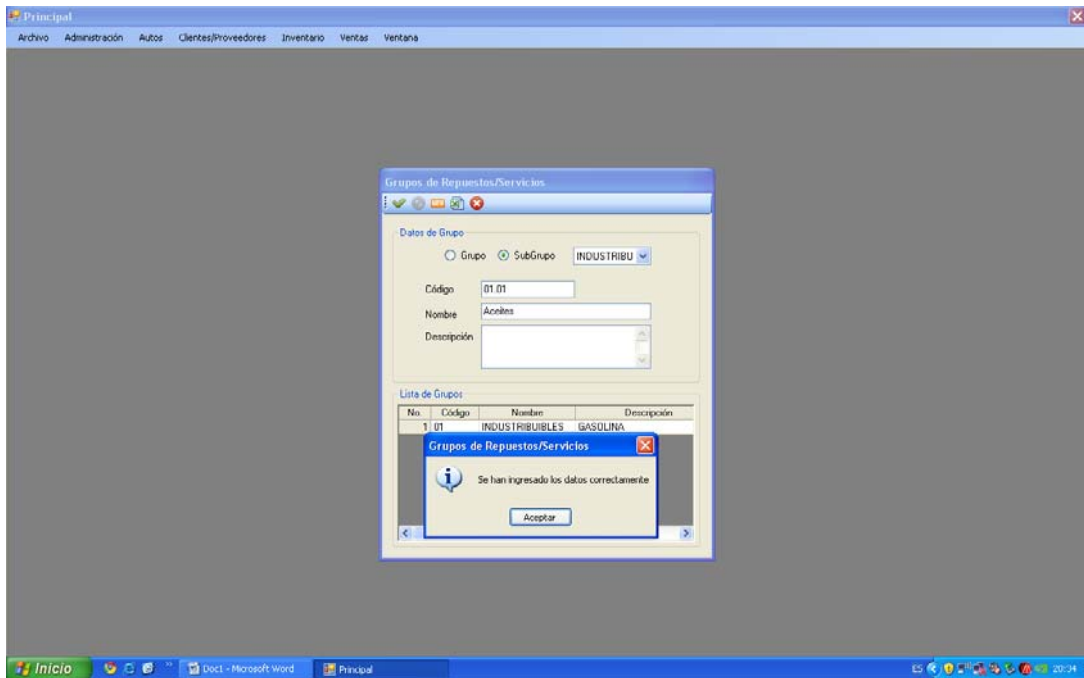


Gráfico 3.69. Creación sub grupo

30) Creación grupo materiales

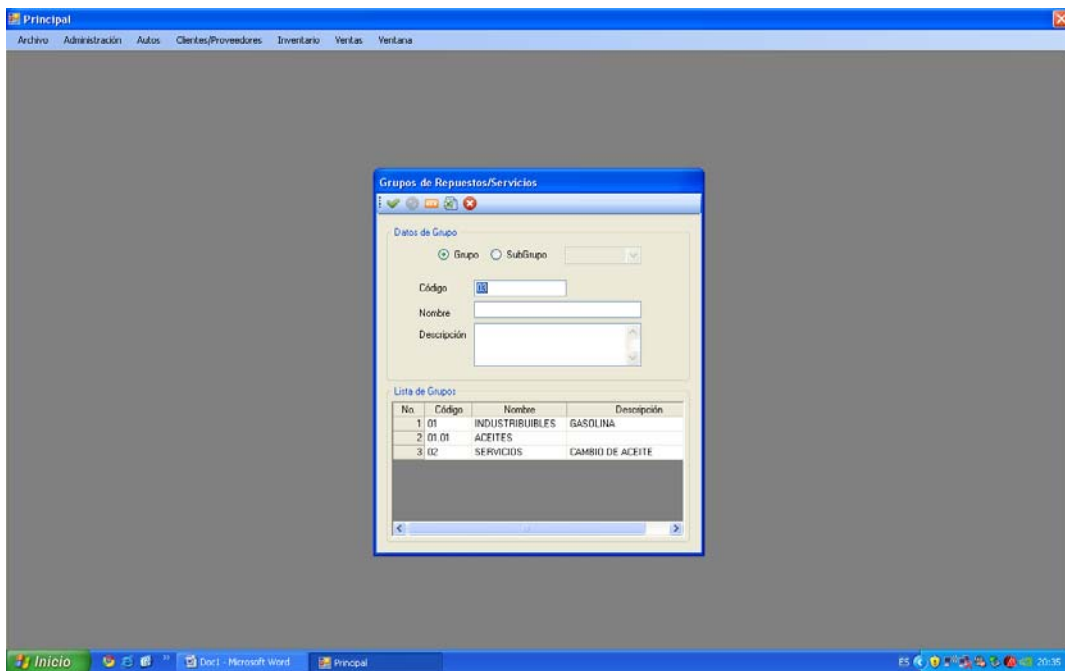


Gráfico 3.70. Creación grupo materiales

31) Registro información de repuestos

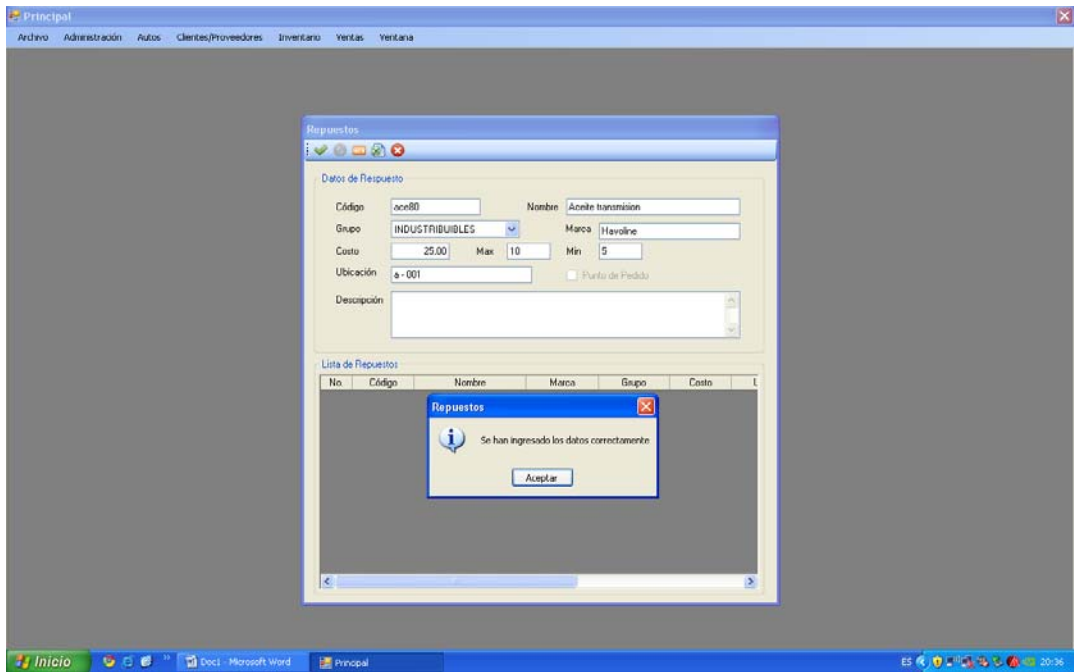


Gráfico 3.71. Registro información de repuestos

32) Registro información servicios

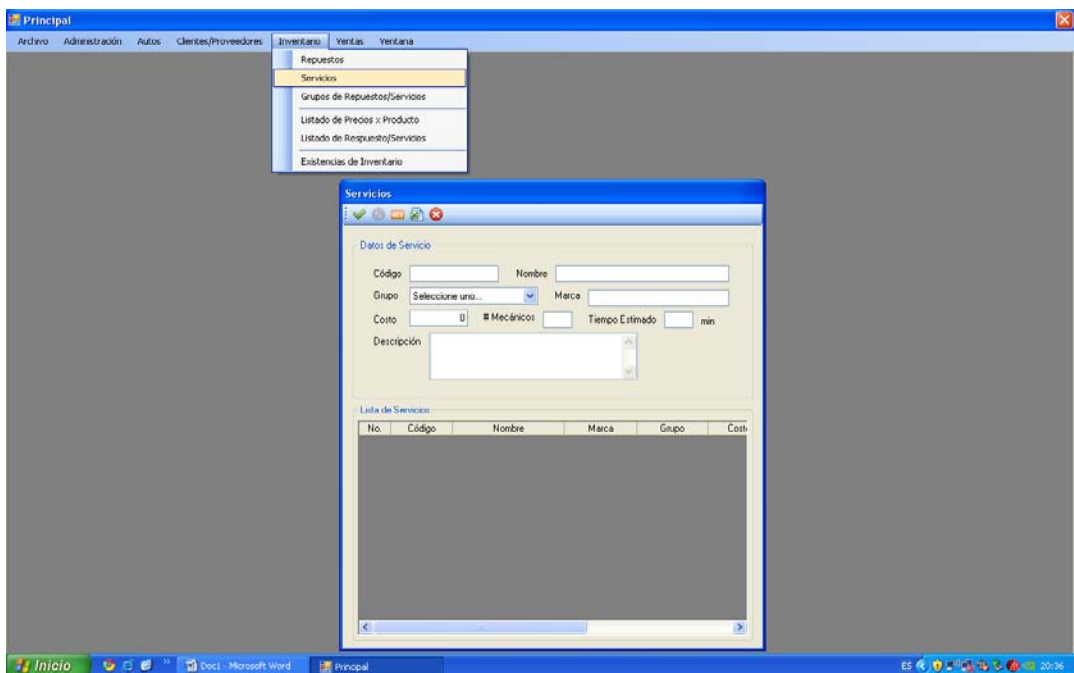


Gráfico 3.72. Registro información servicios

33) Confirmación registro información servicios

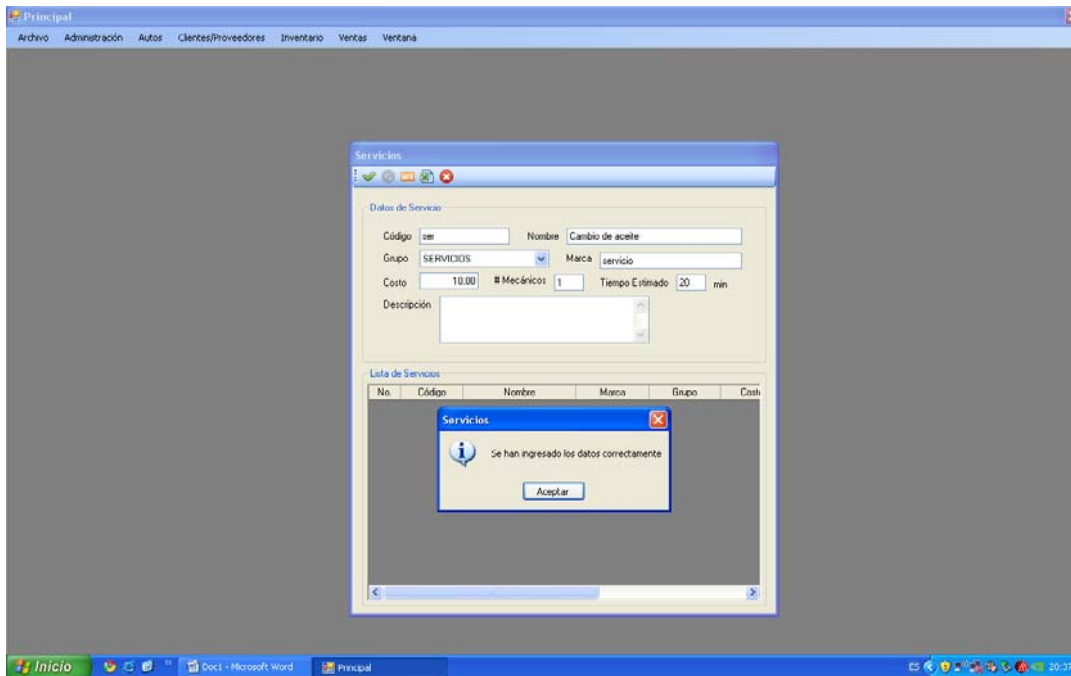


Gráfico 3.73. Confirmación registro información servicios

34) Generación proforma

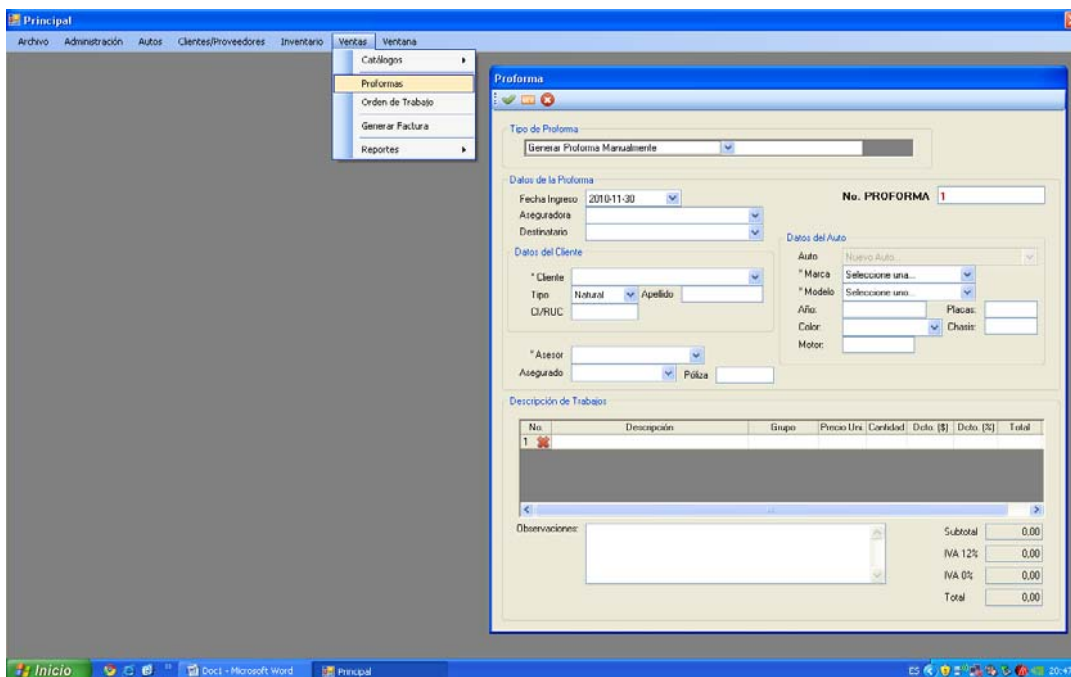


Gráfico 3.74. Generación proforma

35) Escoger generar proforma manualmente

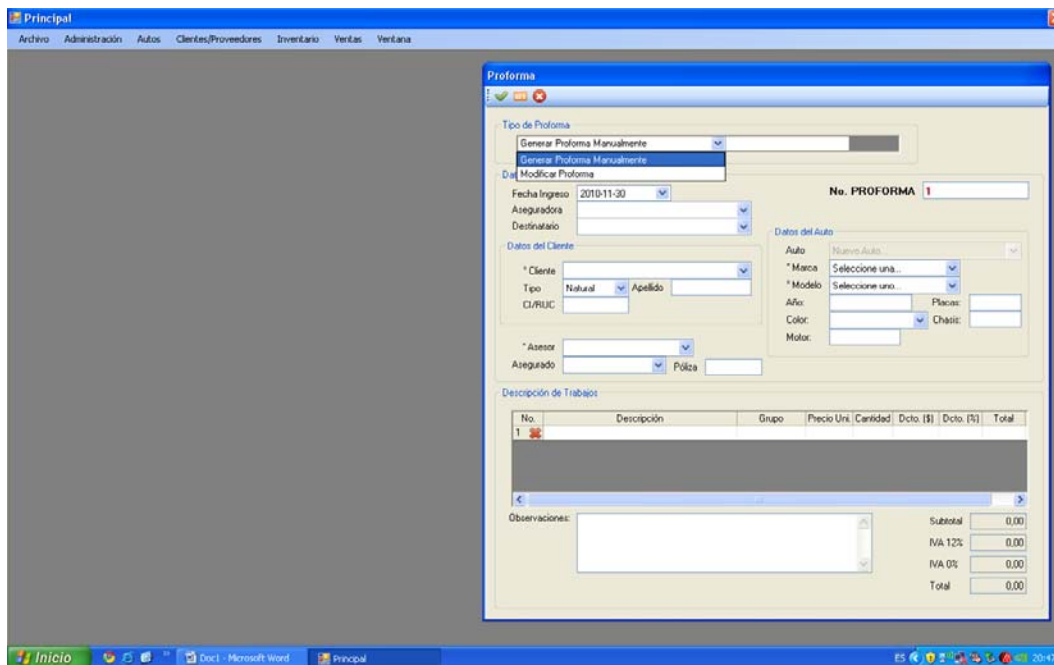


Gráfico 3.75. Proforma manualmente

36) Confirmación generación proforma

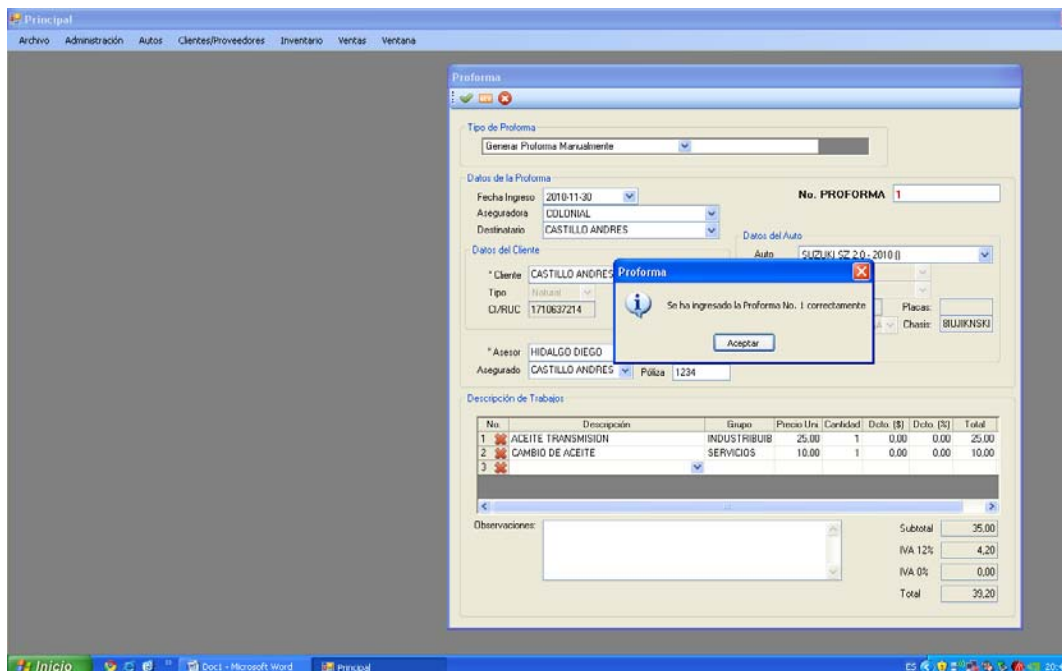


Gráfico 3.76. Confirmación generación proforma

37) Generación orden de trabajo

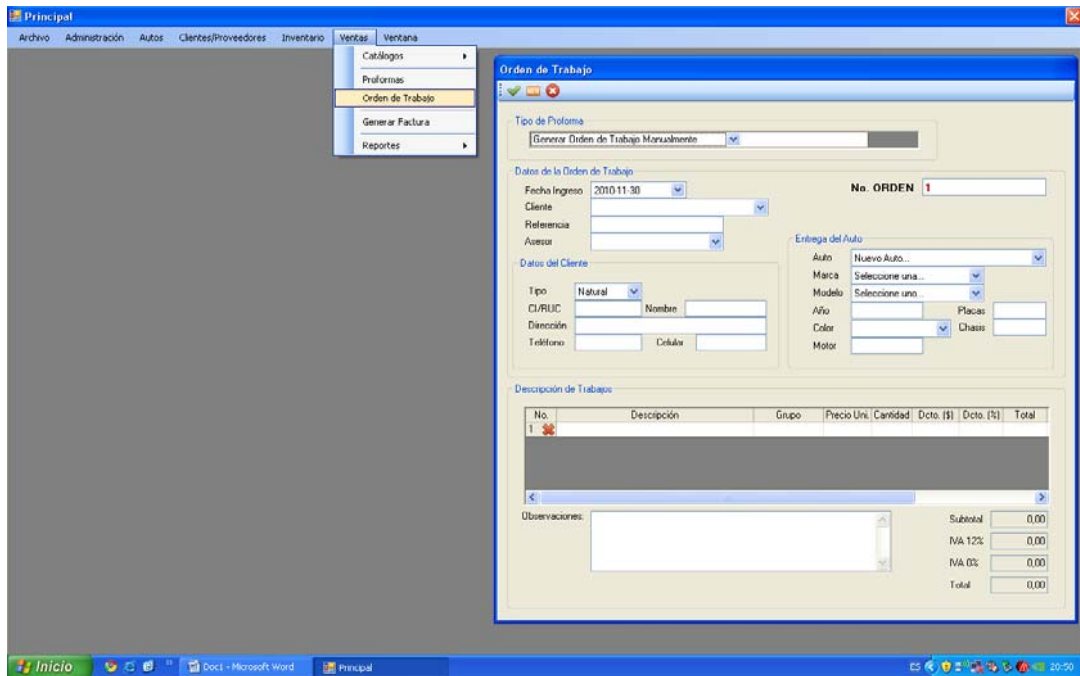


Gráfico 3.77. Generación orden de trabajo

38) Confirmación grabado orden de trabajo

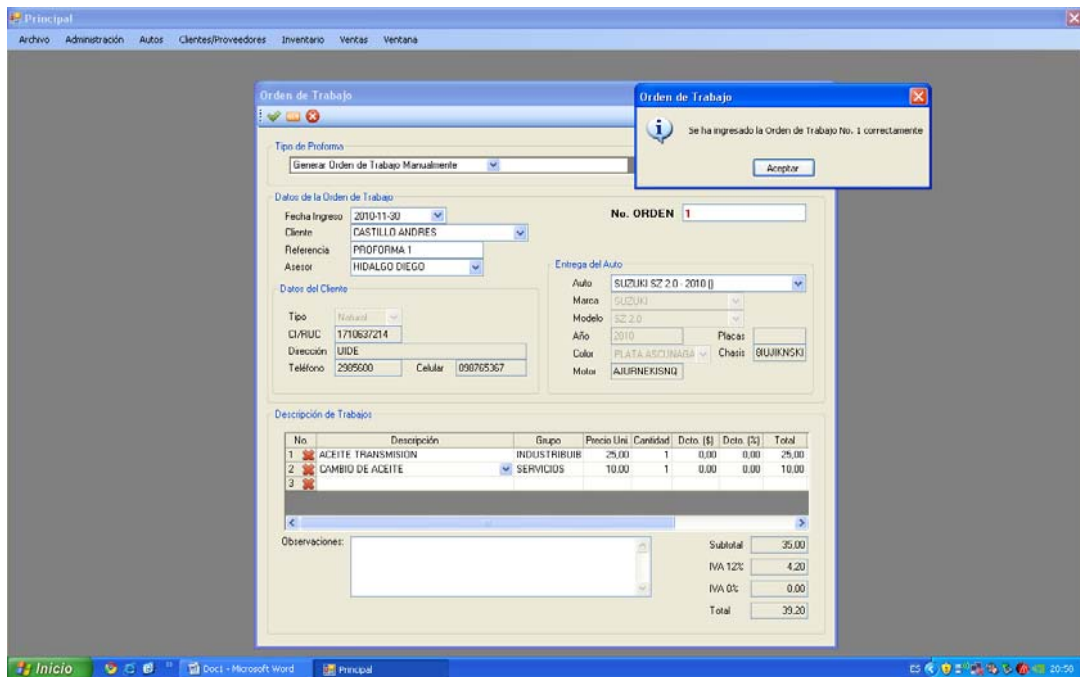


Gráfico 3.78. Confirmación grabado orden de trabajo

39) Dada de baja en inventario de acuerdo a orden de trabajo

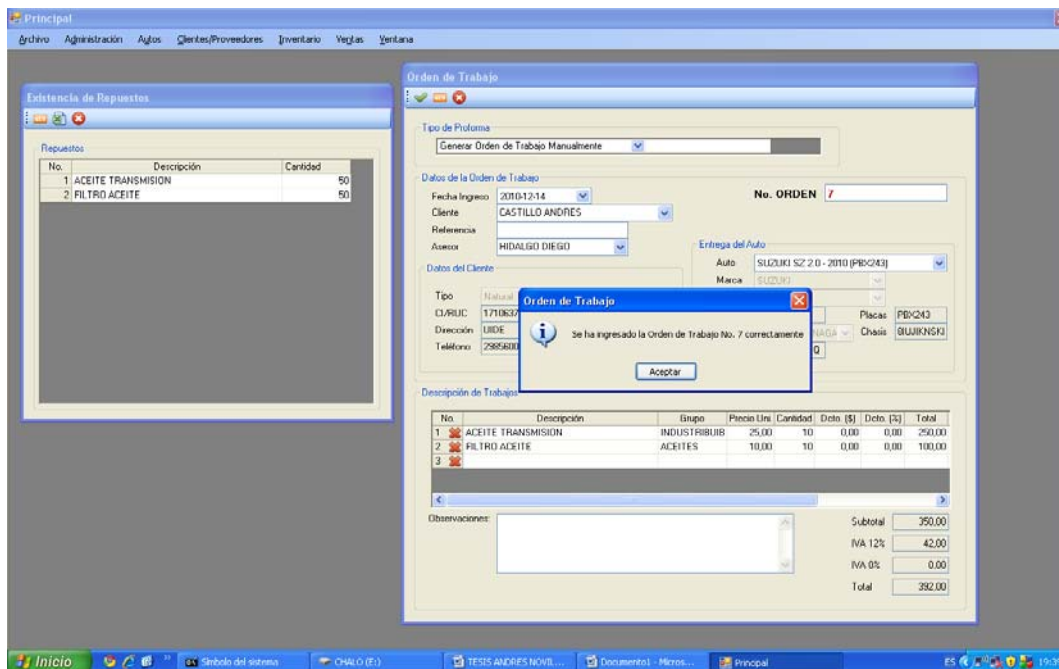


Gráfico 3.79. Dada de baja en inventario de acuerdo a orden de trabajo

40) Nuevo inventario

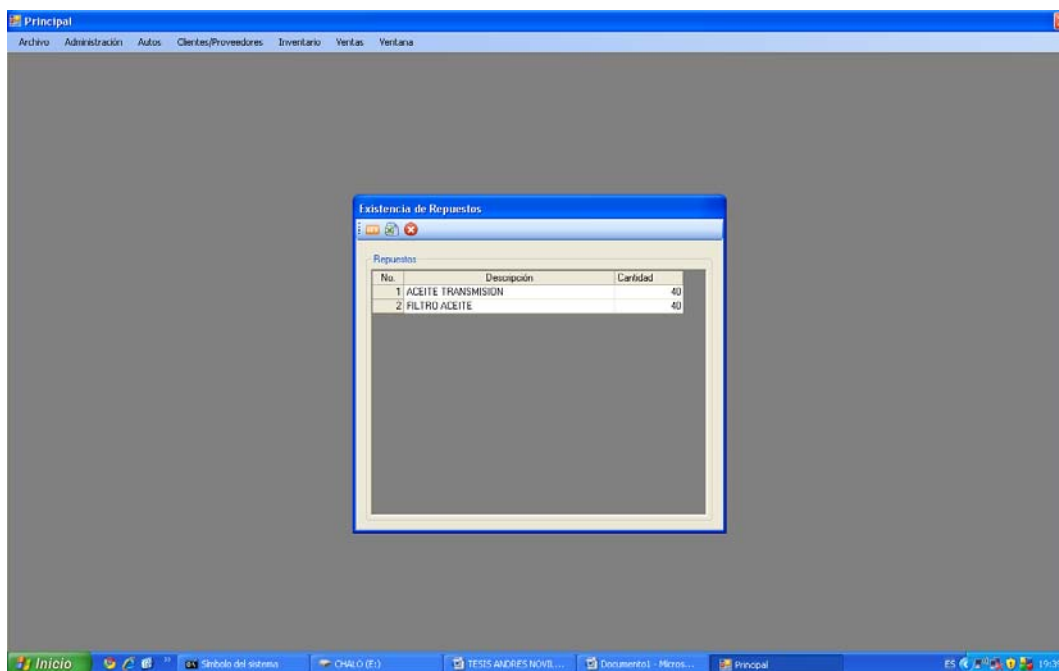


Gráfico 3.80. Nuevo Inventario

41) Mensaje de alerta cuando inventario no es suficiente

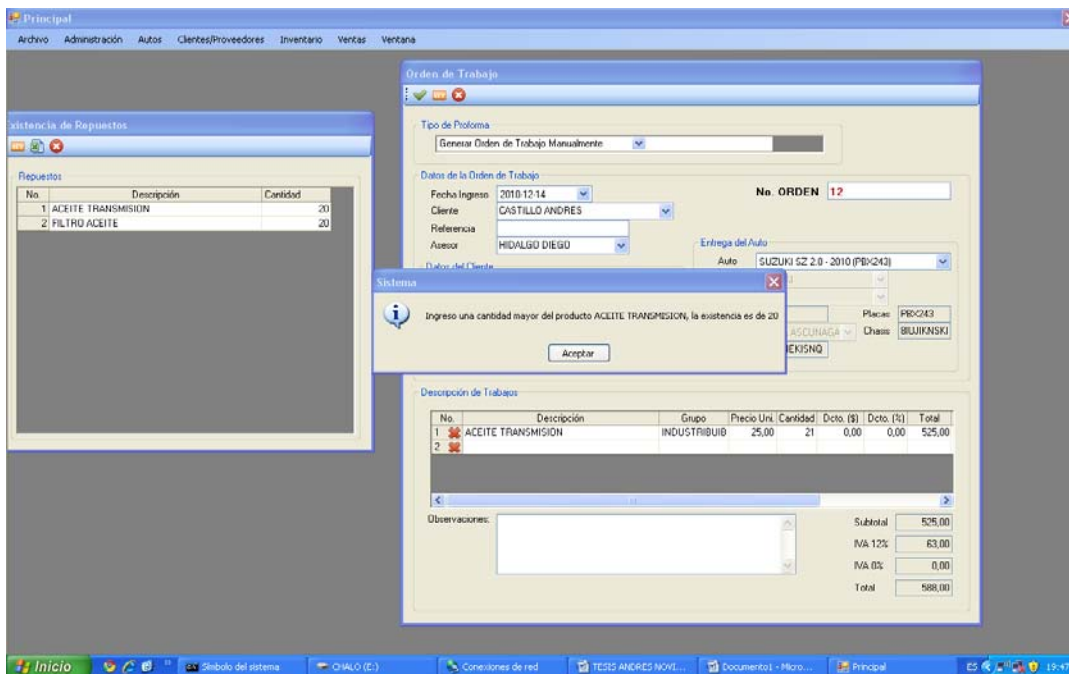


Gráfico 3.81. Mensaje de alerta cuando inventario no es suficiente

42) Registro condición de pago

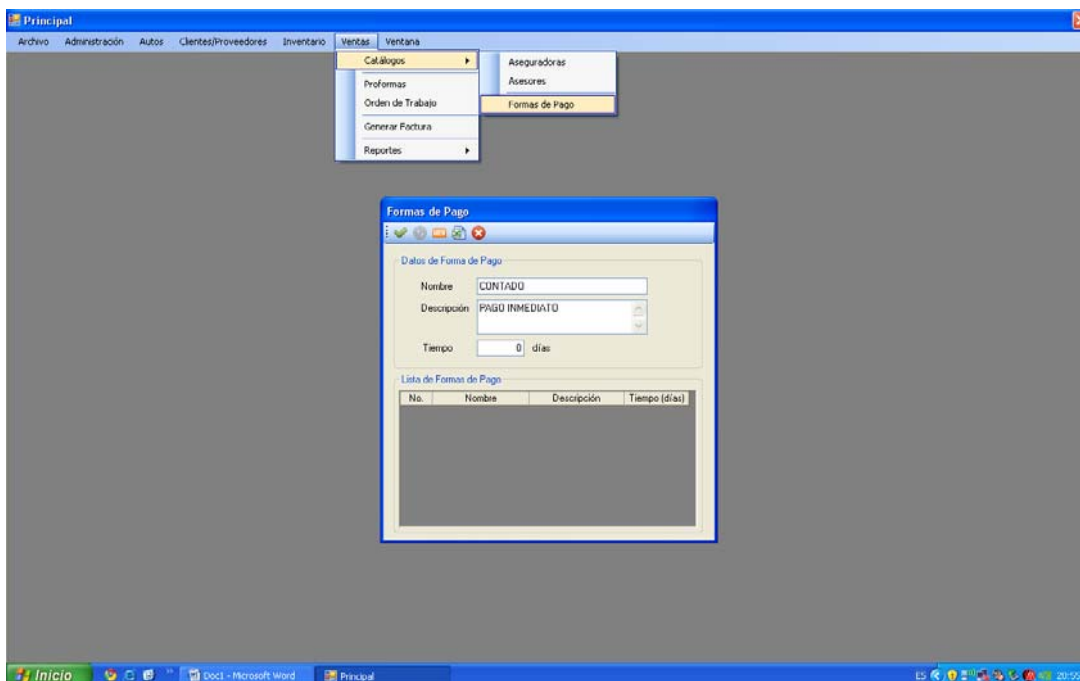


Gráfico 3.82. Registro condición de pago

43) Generar factura

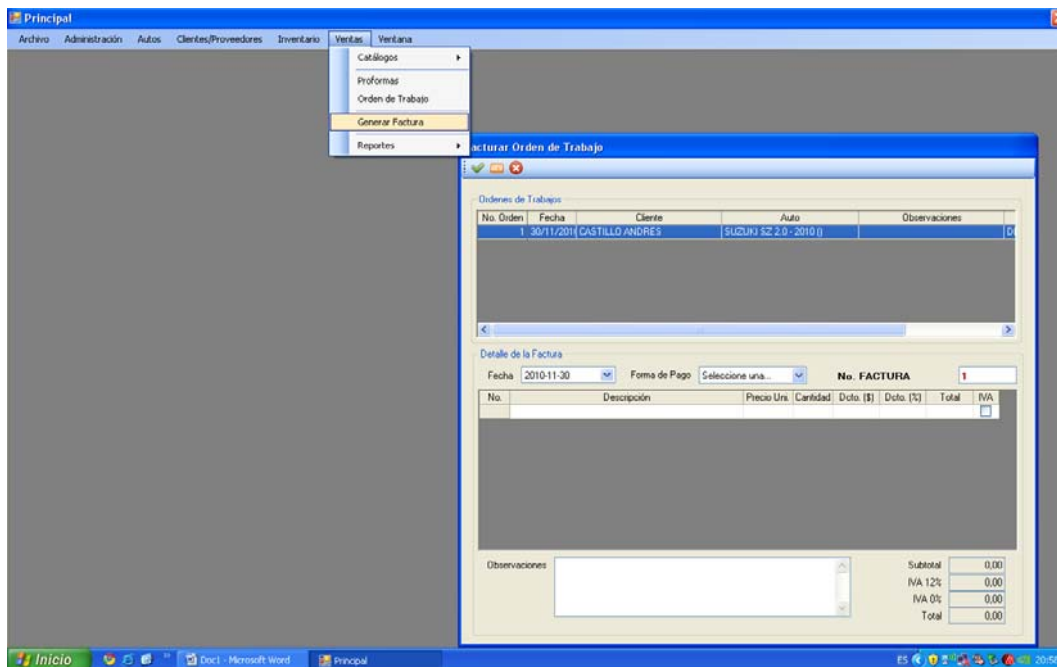


Gráfico 3.83. Generar factura

44) Seleccionar generar factura de orden de trabajo

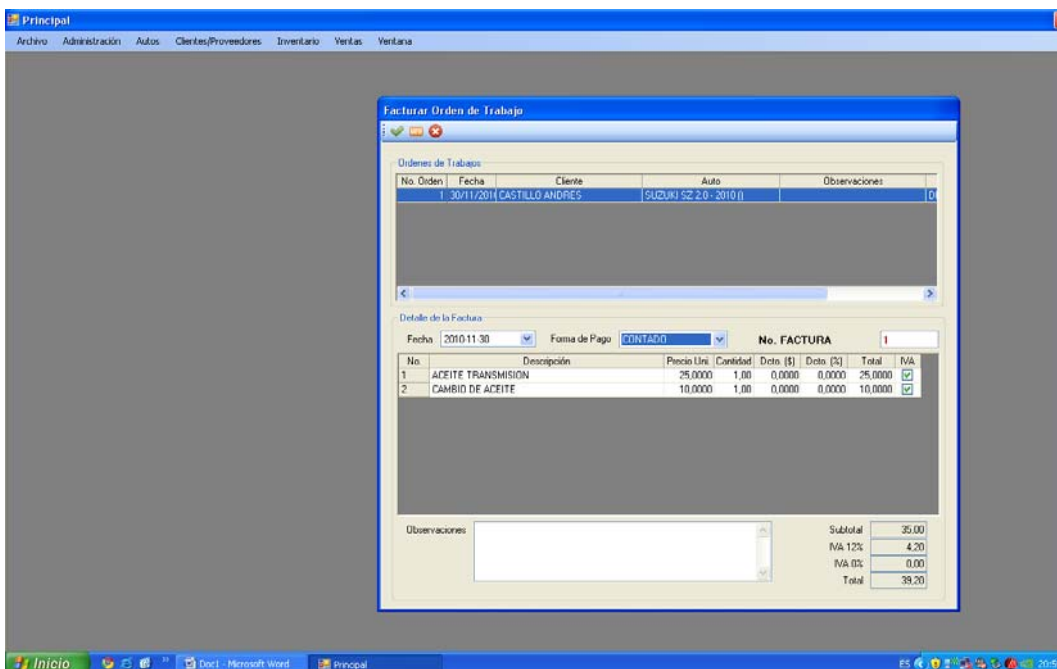


Gráfico 3.84. Facturar a partir de OT

45) Confirmación de grabado factura

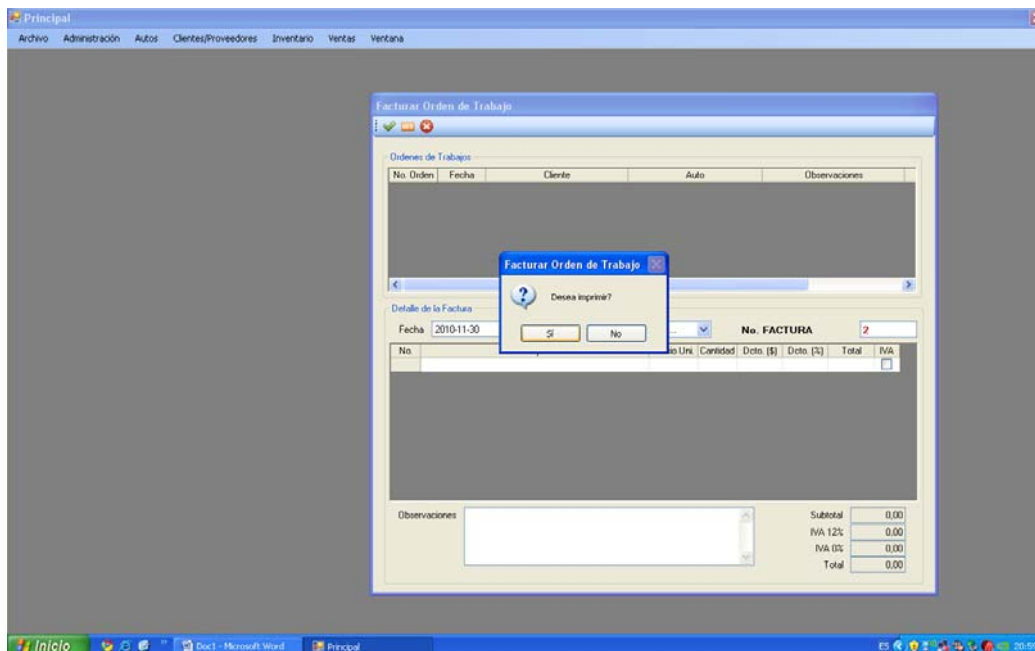


Gráfico 3.85. Confirmación de grabado factura

46) Bajar informes órdenes de trabajo realizadas

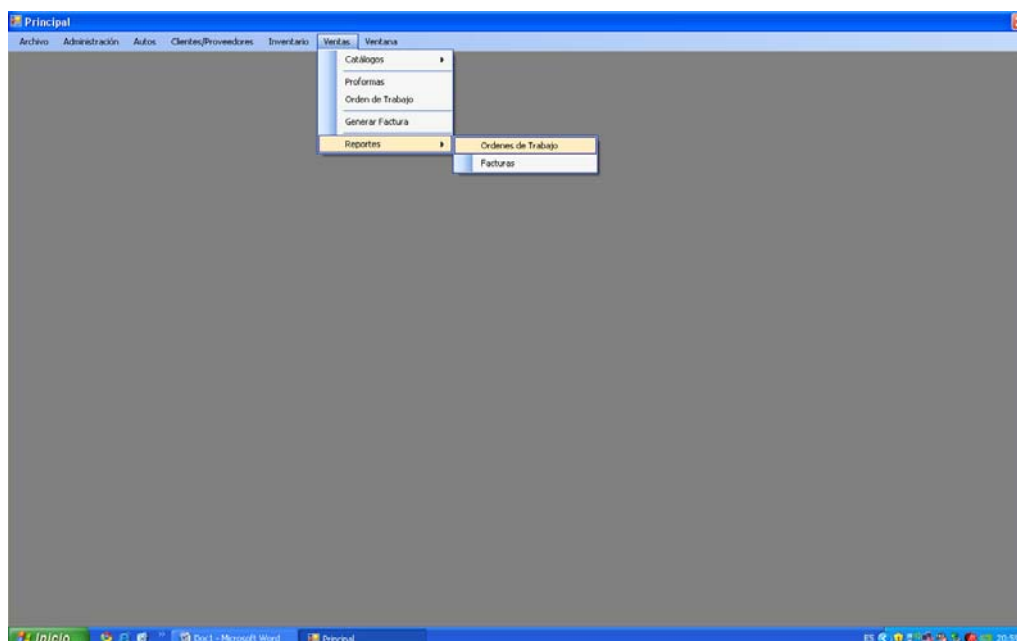
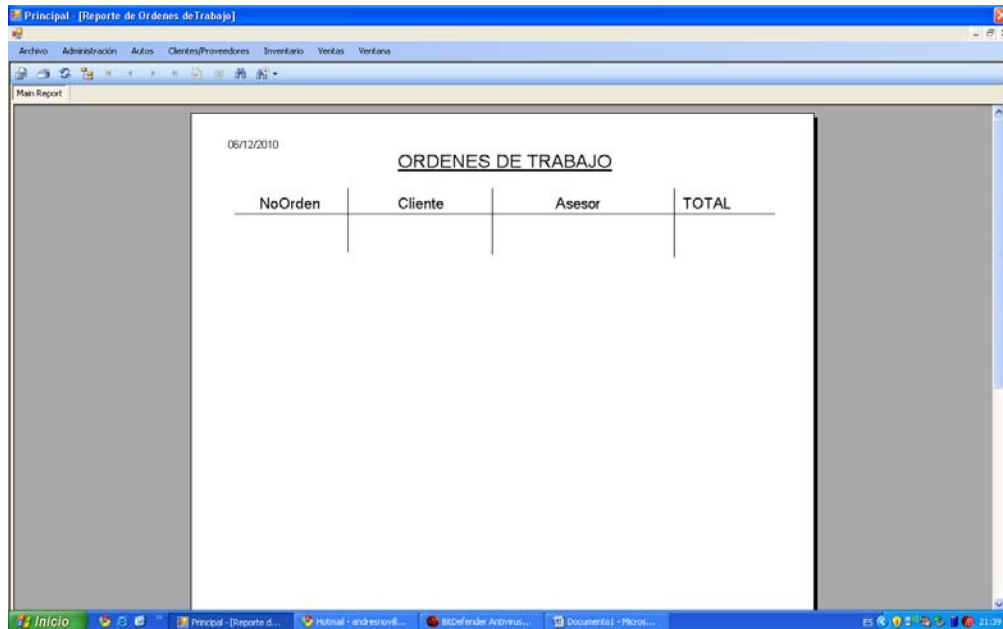


Gráfico 3.86. Informes órdenes de trabajo

47) Reporte orden de trabajo



The screenshot shows a software window with a menu bar (Archivo, Administración, Ayudas, Clientes/Proveedores, Inventario, Ventas, Ventanas) and a toolbar. The main area displays a report header with the date '06/12/2010' and the title 'ORDENES DE TRABAJO'. Below the title is a table with four columns: 'NoOrden', 'Cliente', 'Asesor', and 'TOTAL'. The table is currently empty.

NoOrden	Cliente	Asesor	TOTAL
---------	---------	--------	-------

Gráfico 3.87. Reporte orden de trabajo

48) Generar reporte de facturas

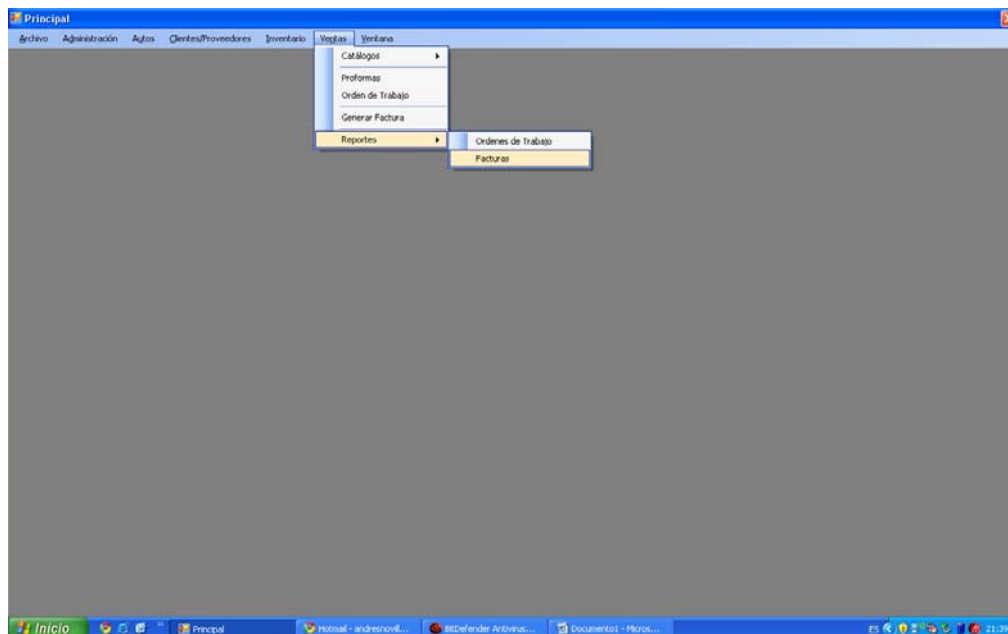


Gráfico 3.88. Generar reporte de facturas

49) Reporte de factura

Principal - [Reporte de Facturación]

Archivo Administración Autos Clientes/Proveedores Inventario Ventas Ventana

Man Report Seleccione el No. de Factura Ingresada

Fecha:	FACTURA	No. de Orden:	
Cliente:	Siniestro:	Placa:	
Referencia:	Marca:	Año:	
Asesor:	Modelo:		
Repuesto/Servicio	Precio	Cantidad	Observaciones
			IVA:
			TOTAL:

Inicio Principal - [Reporte d... Mutual - andresov@... Defensor Antivirus... Documento1 - Micro...

Gráfico 3.89. Reporte de factura

CAPÍTULO 4

Mejoras propuestas

4.1. MEJORA DE ACTIVIDADES EXISTENTES

Con la ayuda del programa existirá el control de los movimientos ocurridos dentro de taller, evitando exceso de movimientos administrativos, robos y malentendidos. Las mejoras que podríamos obtener serian las siguientes:

a.- Control sobre todos los pasos de la administración del taller.- con la implementación del software, toda información del día a día estará centralizada, y almacenada en el mismo lugar. La opción de bajar informes en formato Excel también contribuirá al control.

b.- Control sobre recursos humanos.- Al tener la información de costos, y tiempos para cada servicio, la dirección del taller tendrá un mejor control de efectividad del personal

c.- Saber cuándo se recibió y cuando se entregó una orden de servicio y por cuál de los empleados.- controlar esta parte critica del servicio es de suma importancia para dar el seguimiento adecuado a cualquier problema de calidad o de baja satisfacción del cliente

d.- Dar presupuestos de manera eficiente.- El software al manejar el catálogo completo de servicios y partes de reposición, con sus respectivos precios, permite una exacta forma de realizar presupuestos. Esta forma exacta incluye uniformidad de precios, estandarización de precios por tipo de servicios.

e.- Mejorar el tiempo de entrega de servicios.- La estandarización de tiempos de servicios, ayuda al jefe de taller a planificar y controlar de mejor manera a su recurso humano, con lo cual los tiempos de entrega de servicios mejoraran significativamente

f.- Control sobre todos los repuestos.- Parte importante del negocio que significa tener un taller automotriz, es la venta repuestos, conocer a ubicación de los mismos dentro de la bodega principal así como su existencia nos ayudara a entregar al cliente final siempre un servicio completo que incluya la dotación de repuestos

h.- Tener siempre suficiente abasto de piezas en el inventario.- El software propone un control de inventario basado en el monitoreo de stocks de cada repuesto tomando en cuenta puntos máximos, puntos mínimos y puntos de pedido. Con la opción de bajar informes de todo el inventario de la bodega donde se especifica cuales materiales están entrando en puntos de pedido se puede comenzar con el proceso de compras.

i.- Reporte de ventas diarias, semanales, mensuales y anuales.- Reportes de ventas nos ayudan a entender como marcha el negocio, y poder tomar las acciones respectivas de darse el caso que los números estén lejos de las metas propuestas por la dirección, o por el contrario entender que es lo que se está haciendo bien, para que las ventas estén mejorando, y mejorar esta actividad.

q.- Evitar malos entendidos con el personal y con los clientes.- Una ayuda grande a la hora de resolver un conflicto, es haber tenido todos los acuerdos previo al trabajo bien definidos y aceptados por ambas partes

k.- Tener a los clientes satisfechos, el orden, la reducción de tiempos y la continua mejora a la calidad de los servicios siempre serán un motivo para que los clientes regresen

l.- Incrementar las ventas

CAPÍTULO 5

Conclusiones y recomendaciones.

5.1. Conclusiones

La correcta administración muchas veces olvidada por el trajín de diario, debe ser tomada en cuenta con la misma importancia que se da a las actividades diarias. Esto suele ser el talón de Aquiles de muchos negocios que podrían llegar a ser exitosos.

El uso de herramientas informáticas como soporte para el quehacer diario es de gran ayuda, tecnifica, controla y estandariza muchas cosas que día a día hacemos.

5.2. Recomendaciones

Los procesos dentro de una organización tienen que estar claramente detallados, los cuales deben ser conocidos por todos los miembros de la misma. El cumplimiento diario de los mismos debe ser auditado y documentado.

Al momento de plantear nuevos procesos, se debe buscar la estandarización de las actividades diarias, pero a la vez la simplificación de los mismos. La optimización de los recursos tiene que ser una prioridad dentro de estos nuevos procesos.

Una interfase amigable, fácil de entender y con ayudas visuales siempre facilitara el entendimiento y el “romance” entre el software y los empleados de

cualquier organización. La capacitación es de vital importancia, ya que una fuerza laboral sin la capacitación necesaria podría resultar en que la herramienta se este sub utilizando, y la inversión sería inútil.

La mejora continua, necesita de datos claros y concisos para una adecuada toma de decisiones

La tecnología informática esta en constante evolución, y avanza a pasos mas rápidos que cualquier otra ciencia, razón por la cual se entregan todos los códigos de programación para cualquier mejora que se desee realizar al software.

BIBLIOGRAFÍA

Gestiopolis. Qué es un perfil ocupacional y cómo se construye?

<<http://www.gestiopolis.com/recursos/experto/catsexp/pagans/rh/16/perfilocupacional.htm>> [Consulta: 12 julio 2010]

GRYNA, Frank. Análisis y planeación de la calidad. México: Mcgraw-hill, 1998. p:1-225

Ministerio de Fomento. La gestión por procesos.

<<http://www.fomento.es/NR/rdonlyres/9541acde-55bf-4f01-b8fa-03269d1ed94d/19421/CaptuloIVPrincipiosdelagestindelaCalidad.pdf>> [Consulta: 20 septiembre 2009]

Proceso de Compra <<http://sistemas.itlp.edu.mx/tutoriales/mercadotecnia1/t43.htm>> [Consulta: 12 julio 2010]

Red Hat Enterprise Linux 4. Introducción a la administración de sistemas.

<<http://www.gb.nrao.edu/pubcomputing/redhatELWS4/RH-DOCS/rhel-isa-es-4/ch-intro.html>> [Consulta: 12 julio 2010]

Wikipedia. Microsoft Visual Studio.

<http://es.wikipedia.org/wiki/Microsoft_Visual_Studio> [Consulta: 20 septiembre 2009]

ANEXOS

Anexo 1

Norma ISO

"Documentos" (control/gestión de la documentación).

El establecimiento de un sistema de control de documentos efectivo es un requisito básico para cualquier empresa si su sistema de gestión va a ser certificado contra los requisitos relevantes de ISO/HACCP. En todos los estándares - ISO 9000:2001, 14001:2004, OHSAS 18001:1999, ISO 22000:2005 (HACCP) - este requisito es claramente establecido:

ISO 9001:2008:

Capítulo 4.1.: "La organización debe establecer, documentar, implementar y mantener un "sistema de gestión de calidad".

Capítulo 4.2.3.: "Los documentos requeridos por el sistema de gestión de calidad deben ser controlados".

En otras palabras, implementar y mantener un sistema de gestión de documentos efectivo es un pre requisito "obligatorio" para la existencia de un Sistema de Gestión de Calidad exitoso. El Software es la herramienta que ayudará a lograrlo.

El Software tiene un enfoque para el desarrollo del sistema de control de documentos - es relevante, simple y eficaz. Todo lo que se requiere es que el usuario ingrese la información principal acerca de los documentos en la base de datos, y luego el usuario procederá a bajar informes a una hoja de cálculo

Por otro lado, se puede utilizar el Software como una herramienta de control de documentos, no solo aquellos incluidos en el Sistema de Gestión (Calidad, SGA etc.)

De esta forma El Software permitirá ser capaz de desarrollar e implementar un sistema eficiente de gestión y control de documentos de conformidad con los estándares ISO. .

El sistema de generación de reportes según los resultados de actos realizados permite también almacenarlos en copia dura o electrónica y utilizarlos así como las anotaciones realizadas del Sistema.

La organización de acceso de usuarios a los archivos de documentos del Sistema conforme a su nivel de autorización ("acceso con contraseña").

El Software nos proporcionará la posibilidad de utilizar el programa como un instrumento de control para la circulación de los documentos de la administración del taller, y no sólo para aquellos que se refieren al sistema de gestión.

"Procesos" (descripción/gestión de procesos, enfoque de proceso).

El estándar ISO 9000:2000 promueve la adopción de un enfoque de proceso (en otras palabras - la descripción de procesos y el sistema de gestión empresarial orientada a los procesos) en el desarrollo, implementación y mejoramiento de la calidad del sistema de gestión (ambiente, seguridad alimentaria). El enfoque de procesos es reflejado en la estructura de ISO 9004:2000 (Sistemas de Gestión de la Calidad -Guías para el mejoramiento de desempeño), y también en ISO 9001:2008 (Sistemas de Gestión de la Calidad-Requisitos).

Para implementar el enfoque de procesos, es necesario conocer cuáles son los requisitos necesarios para dar cumplimiento y el valor de cumplirlos tanto para el taller como para el cliente.

a.- El Software es una herramienta relevante, simple, económica y efectiva para el desarrollo, implementación y adaptación del enfoque de procesos en su sistema de gestión de calidad en el taller.

"Capacitación" (desarrollo y gestión de capacitación del personal).

"La organización debe:

a.- Determinar la competencia necesaria para el personal que realizando el trabajo influye en la calidad del producto.

b.- Proporcionar la capacitación o tomar las acciones correspondientes para satisfacer las necesidades de capacitación.

c.- Evaluar la eficacia de las acciones tomadas.

d.- Asegurar que su personal sea consciente de la relevancia e importancia de sus actividades y cómo ellas contribuyen al logro de los objetivos de calidad;

e.- Mantener los registros apropiados de la educación, capacitación, aptitudes y experiencia."

"Auditorías" (auditorías / las acciones correctivas/preventivas).

Estándar ISO 9001:2008, capítulo. 8.2:

"La organización conducirá auditorías internas en los intervalos planeados para determinar la calidad del sistema de gestión:

a.- Es conforme a las medidas planeadas, para dar cumplimiento a los requisitos de este estándar internacional y a los requisitos del sistema de gestión de la calidad establecidos por la organización

b.- Sea efectivamente implementado y mantenido."

El programa de auditoría debe ser planeado tomando en consideración el estado y la importancia de los procesos y las áreas sujetas a la auditoría. Deben definirse los criterios así como el alcance y la frecuencia de las auditorías. La

selección de los auditores y ejecución de auditorías deben asegurar la objetividad e imparcialidad de los procesos de auditoría. Los auditores no deben auditar su propio trabajo.

De manera general, las auditorías de calidad tienen como fin lograr los siguientes objetivos:

a.- Determinar hasta qué punto su sistema de calidad:

- Logra sus objetivos.
- Es conforme a sus requisitos.
- Cumple con los requisitos regulatorios.
- Cumple los requisitos contractuales de sus clientes.
- Es conforme a un estándar de calidad reconocido.

b.- Para mejorar la eficiencia y eficacia de su sistema de calidad.

c.- Para enlistar su sistema de calidad en el registro de una agencia independiente.

d.- Para verificar que su sistema de calidad continúa cumpliendo los requisitos.

e.- Para asegurar que todos los auditores tienen la apropiada capacidad y aptitudes.

. "Quejas" (gestión de solicitudes/quejas de clientes).

Este módulo será su asistente en la implementación de los requisitos del estándar ISO 10002:2004 "Gestión de Calidad. Satisfacción de clientes. Guías para el manejo de quejas en las organizaciones" y el estándar ISO 9001:2008, Capítulo 7.2.3. "Retroalimentación de Clientes".

Cuando se implementa el proceso de manejo de quejas, el cliente se beneficia del tratamiento responsable de su queja. La organización se beneficia por la

identificación del área problema y de las oportunidades para el mejoramiento y de los ahorros que esto implica.

- La implementación del proceso descrito en este estándar le permite:
- Proporcionar a una queja el acceso a un proceso de manejo de quejas abierto y responsable.
- Mejorar la habilidad de la organización para resolver quejas de una manera consistente, sistemática y responsable para satisfacción de todas las partes.
- Habilitar a la organización para identificar tendencias y eliminar las causas de las quejas, así como para mejorar las operaciones en la organización.
- Ayudar a la organización a adoptar un sistema enfocado a clientes, para resolver quejas, y animar al personal a mejorar sus habilidades en el trabajo con los clientes.

- Proporcionar una base para la revisión y análisis continuos del proceso de manejo de quejas, la resolución de las mismas, así como procesar los mejoramientos llevados a cabo.

"Des conformidades" (registro y control de no conformidades, análisis comparativo de no conformidades, evaluación cuantitativa de la eficacia del funcionamiento del Sistema de Gestión, representación gráfica (diagramas) de los resultados del análisis).

El estándar 9001:2008, capítulo 8.5.1 "Mejoramiento continuo" requiere: " La organización debe mejorar de manera continua la eficacia del sistema de gestión de la calidad utilizando una política de calidad, objetivos de calidad, resultados de auditoría, análisis de datos, acciones correctivas y preventivas y revisión por la Dirección".

Capítulo. 5.6 "Revisión por la Dirección" establece: "La alta dirección deben revisar el sistema de gestión de calidad de la organización, en intervalos planeados, para asegurar su continua eficacia" - en otras palabras, la eficacia del sistema de gestión debe ser revisada/evaluada en una base regular.

El Módulo "No conformidades" le ayudará a cumplir estos y otros requisitos de los estándares ISO:

- Desarrollar un sistema para la clasificación de no conformidades.
- Definir los criterios de evaluación cuantitativa.
- Organizar el registro de no conformidades y llevar un control efectivo sobre su eliminación en el transcurso del tiempo.
- Conducir el análisis comparativo de no conformidades basado en varios parámetros para ajustarse a sus necesidades en particular.
- Ejecutar el análisis cuantitativo de la eficacia de la implementación y funcionamiento de su sistema de gestión en su organización.
- Resumir y presentar los resultados del análisis en una forma vívida (gráfica) y adecuada para las necesidades del análisis de la gestión y para la toma de decisiones de la empresa.

Anexo 2

Código Fuente

Clase de Conexión

```
Imports System.Data
Imports System.Data.Odbc
Imports System.EnterpriseServices
Imports System.Data.SqlClient

Public Class clsConexion

    Public Function Conectar(ByVal User As String, ByVal Clave As String)
    As Boolean
        Dim cadenaM As String
        cadenaM = "driver={MySQL ODBC 3.51 Driver};" _
            & "server=" & strServidorBDDM & ";" _
            & "uid=" & User & ";" _
            & "pwd=" & Clave & ";" _
            & "database=" & strBDD & ";" _
            & "port= " & strPuertoM & ";" _
            & "OPTION=1;"

        ConnMaster = New OdbcConnection(cadenaM)
        ConnMaster.ConnectionTimeout = 30

        Try
            ConnMaster.Open()
            Return True
        Catch ex As Exception
            MsgBox(ex.Message)
            Return False
        End Try
    End Function

    Public Sub Desconectar()
        ConnMaster.Close()
    End Sub
End Class
```

Clase de SQL

```
Imports System.Data
Imports System.Data.Odbc

Public Class clsSQL
    Private OdbcCommandM As OdbcCommand
    Public dataR As OdbcDataReader ' ADODB.Recordset
    Private OdbcConnectionM As OdbcConnection
    Private Servidor As String
    Private strSQ As String
    Public Sub Inicializar(ByVal adocon_ParM As OdbcConnection)
```

```

        OdbcCommandM = New OdbcCommand
        OdbcConnectionM = adocon_ParM
        OdbcCommandM.CommandText = strSQ
        OdbcCommandM.Connection = OdbcConnectionM
    End Sub

    Public Sub Ejecutar(Optional ByVal strSQ As Object = "", Optional ByVal
donde As String = "L", Optional ByVal que As String = "S")
        Servidor = donde
        If donde = "L" Then
            If strSQ <> "" Then
                OdbcCommandM.CommandText = strSQ
            End If

            If que <> "S" Then
                OdbcCommandM.ExecuteNonQuery()
            Else
                dataR = OdbcCommandM.ExecuteReader
            End If
        End If
    End Sub

End Class

```

Inicio de Sesion

```

Imports System.IO
Public Class frmInicioSesion
    Private SQL As New clsSQL
    Private strSQL As String
    Private Sub OK_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
        Dim Conexion As New clsConexion

        If Trim(txtUsuario.Text) = "" Then
            Mensaje(1, Me.Text)
            Exit Sub
        End If
        If Conexion.Conectar("root", "123") = True Then

            SQL.Inicializar(ConnMaster)

            strSQL = " SELECT usu_codigo,usu_clave " & _
                " FROM usuario " & _
                " WHERE usu_codigo='" & Texto(txtUsuario.Text) & "' "
            & _
                " AND usu_clave='" & txtClave.Text & "'"
            SQL.Ejecutar(strSQL)

            If SQL.dataR.HasRows = True Then

                strUsuario = UCase(Trim(txtUsuario.Text))
                strClave = txtClave.Text

                mdiPrincipal.Close()
            End If
        End If
    End Sub
End Class

```



```

        mdiPrincipal.Show()

        Me.Close()

    Else
        SQL.dataR.Close()
        MsgBox("El usuario o la contraseña son incorrectas,
inténtelo nuevamente", MsgBoxStyle.Information, Me.Text)
    End If
    Else
        txtClave.Focus()
    End If

End Sub

Private Sub Cancel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub

Private Sub frmInicioSesion_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
    Dim strVar As String
    Dim strVal As String

    strPathConfig = Application.StartupPath & "\Config.cfg"
    If File.Exists(strPathConfig) = True Then
        Dim sr As New StreamReader(strPathConfig)
        Dim linea As String
        While Not sr.EndOfStream
            linea = sr.ReadLine()
            strVar = LCase(Trim(Microsoft.VisualBasic.Left(linea,
InStr(linea, "=") - 1)))
            strVal = Trim(Microsoft.VisualBasic.Right(linea, Len(linea)
- InStr(linea, "=")))
            If Len(strVal) <> 0 Then
                Select Case strVar
                    Case "servidorbddm"
                        strServidorBDDM = strVal
                    Case "bdd"
                        strBDD = strVal
                    Case "puertom"
                        strPuertoM = strVal
                End Select
            End If
        End While
        sr.Close()
    Else
        MsgBox("Archivo de Configuración", MsgBoxStyle.Information,
Me.Text)
        frmConfiguracion.Show()
        Me.Close()
    End If
End Sub

```

```

    Private Sub txtUsuario_KeyDown(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyEventArgs) Handles txtUsuario.KeyDown
        If e.KeyCode = Keys.F2 Then
            frmConfiguracion.Show()
            Me.Close()
        End If
    End Sub
End Class

```

Configuracion

```

Imports System.IO
Public Class frmConfiguracion

    Private Sub frmConfiguracion_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
        Dim strVar As String, strVal As String

        If File.Exists(strPathConfig) = True Then
            Dim sr As New StreamReader(strPathConfig)
            Dim linea As String

            While Not sr.EndOfStream
                linea = sr.ReadLine()
                strVar = LCase(Trim(Microsoft.VisualBasic.Left(linea,
InStr(linea, "=") - 1)))
                strVal = Trim(Microsoft.VisualBasic.Right(linea, Len(linea)
- InStr(linea, "=")))
                If Len(strVal) <> 0 Then
                    Select Case strVar
                        Case "servidorbddm"
                            txtServidorM.Text = strVal
                        Case "bdd"
                            txtNombreBDD.Text = strVal
                        Case "puertom"
                            txtPuertoM.Text = strVal
                    End Select
                End If
            End While
            sr.Close()
        End If
    End Sub

    Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
        Dim strLinea As String

        If Trim(txtServidorM.Text) <> "" And Trim(txtNombreBDD.Text) <> ""
Then
            Dim sw As New StreamWriter(strPathConfig)
            strLinea = "servidorbddm = " & Trim(txtServidorM.Text)
            sw.WriteLine(strLinea)
            strLinea = "bdd = " & Trim(txtNombreBDD.Text)
            sw.WriteLine(strLinea)
            strLinea = "puertom = " & Trim(txtPuertoM.Text)

```

```

        sw.WriteLine(strLinea)
        sw.Close()
        frmInicioSesion.Show()
        Me.Close()
    Else
        MsgBox("Llene los campos de:" & vbNewLine & vbNewLine & _
            "-> Servidor de BDD Master" & vbNewLine & _
            "-> Nombre de la Base de Datos" & vbNewLine,
MsgBoxStyle.Information, Me.Text)
        txtServidorM.Focus()
    End If
End Sub

Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    If File.Exists(strPathConfig) = True Then
        frmInicioSesion.Show()
        Me.Close()
    Else
        Me.Close()
    End If
End Sub
End Class

```

Principal

Option Explicit On

Imports System.Windows.Forms

```

Public Class mdiPrincipal
    Private strSQL As String
    Private SQL As New clsSQL
    Private ht As New Hashtable
    Private codigos(10) As String
    Private CodigoMenu As String
    Private ElMenu As String

    Private Sub ExitToolsStripMenuItem_Click(ByVal sender As Object, ByVal
e As EventArgs)
        Me.Close()
    End Sub

    Private m_ChildFormNumber As Integer

    Private Sub mdiPrincipal_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        limpiarMenu()
        CrearMenu(0)
    End Sub

    Public Sub CrearMenu(Optional ByVal tipo As Integer = 0)

        Dim strMenu As String

```

```

strSQL = ""
strMenu = ""
If tipo = 0 Then
    strSQL = " SELECT distinct menu.men_nombrado " & _
            " FROM menu " & _
            " INNER JOIN grupo_permiso " & _
            " ON menu.men_codigo=grupo_permiso.men_codigo " & _
            " INNER JOIN grupo_usuario " & _
            " ON grupo_usuario.gru_codigo=grupo_permiso.gru_codigo
" & _
            " WHERE men_plataforma='W' " & _
            " AND grupo_usuario.usu_codigo= '" & strUsuario & "' "
& _
            " UNION "
End If

strSQL = strSQL & " SELECT distinct men_nombrado " & _
            " FROM menu " & _
            " WHERE men_plataforma='W' " & _
            " AND men_obligatorio='1' "
SQL.Ejecutar(strSQL)

Me.menuPrincipal.Visible = True

ht.Clear()
While SQL.dataR.Read
    ht.Add(SQL.dataR.GetValue(0), SQL.dataR.GetValue(0))
End While
SQL.dataR.Close()

For Each Menu As ToolStripMenuItem In Me.menuPrincipal.Items
    If ht.Contains(Menu.Name) = True Then
        Menu.Visible = True
    End If
    If Menu.DropDownItems.Count > 0 Then
        habilitarMenus(Menu.DropDownItems, strMenu)
    End If
Next
End Sub

Private Sub habilitarMenus(ByVal menus As ToolStripItemCollection,
ByVal nombre As String)
    For Each submenu As ToolStripItem In menus
        If ht.Contains(submenu.Name) = True Then
            submenu.Visible = True
            If CType(submenu, ToolStripMenuItem).DropDownItems.Count >
0 Then
                habilitarMenus(CType(submenu,
ToolStripMenuItem).DropDownItems, nombre)
            End If
        End If
    Next
End Sub

Private Sub limpiarMenu()
    For Each Menu As ToolStripMenuItem In Me.menuPrincipal.Items

```

```

        Menu.Visible = False
        If Menu.DropDownItems.Count > 0 Then
            limpiarMenus(Menu.DropDownItems)
        End If
    Next Menu
End Sub

Private Sub limpiarMenus(ByVal menus As ToolStripItemCollection)
    For Each subMenu As ToolStripItem In menus
        If InStr(LCase(subMenu.Name), "linea") = 0 Then
            subMenu.Visible = False
            If CType(subMenu, ToolStripMenuItem).DropDownItems.Count >
0 Then
                limpiarMenus(CType(subMenu,
ToolStripMenuItem).DropDownItems)
            End If
        End If
    Next
End Sub

Private Sub crearMenus(ByVal menus As ToolStripItemCollection, ByRef
tipo As Integer)
    For Each subMenu As ToolStripItem In menus
        If InStr(LCase(subMenu.Name), "linea") = 0 Then
            tipo = tipo + 1
            CrearMenu(subMenu, tipo)
            If CType(subMenu, ToolStripMenuItem).DropDownItems.Count >
0 Then
                crearMenus(CType(subMenu,
ToolStripMenuItem).DropDownItems, tipo)
            Else
                tipo = tipo - 1
            End If
        End If
    Next
    tipo = tipo - 1
End Sub

Private Sub crearMenu(ByVal Menu As ToolStripMenuItem, ByRef Tipo As
Integer)
    Dim i As Long
    If InStr(Menu.Name, "linea") = 0 Then
        codigos(Tipo) = Aumentar(codigos(Tipo))

        For i = Tipo + 1 To 10
            codigos(i) = "00"
        Next i
        CodigoMenu = ""
        For i = 0 To Tipo
            CodigoMenu = CodigoMenu & codigos(i)
        Next i
        ElMenu = Replace(Menu.Text, "&", "")
        strSQL = "INSERT INTO menu VALUES ('" & CodigoMenu & "','" &
ElMenu & "','" & Menu.Name & "','W','0') "
        SQL.Ejecutar(strSQL)
    End If
End Sub

```

```

        End If
    End Sub

    Private Sub menuCambiarContrasena_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles menuCambiarContrasena.Click
        frmCambiarClave.ShowDialog()
    End Sub

    Private Sub menuCerrarSesion_Click_1(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles menuCerrarSesion.Click
        strUsuario = ""
        strClave = ""
        For Each childForm As Form In Me.MdiChildren
            childForm.Close()
        Next

        Me.Hide()
        frmInicioSesion.ShowDialog()
    End Sub

    Private Sub menuCascada_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menuCascada.Click
        Me.LayoutMdi(MdiLayout.Cascade)
    End Sub

    Private Sub menuMosaicoVertical_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles menuMosaicoVertical.Click
        Me.LayoutMdi(MdiLayout.TileVertical)
    End Sub

    Private Sub menuMosaicoHorizontal_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles menuMosaicoHorizontal.Click
        Me.LayoutMdi(MdiLayout.TileHorizontal)
    End Sub

    Private Sub menuOrganizarIconos_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles menuOrganizarIconos.Click
        Me.LayoutMdi(MdiLayout.ArrangeIcons)
    End Sub

    Private Sub menuCerrarTodo_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles menuCerrarTodo.Click
        ' Close all child forms of the parent.
        For Each ChildForm As Form In Me.MdiChildren
            ChildForm.Close()
        Next
    End Sub

    Private Sub menuSalir_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menuSalir.Click
        Me.Close()
    End Sub

    Private Sub menuGrupos_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menuGrupos.Click
        If frmGrupos.Visible Then

```

```

        frmGrupos.Activate()
    Else
        frmGrupos.MdiParent = Me
        frmGrupos.Show()
    End If
End Sub

Private Sub menuUsuarios_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles menuUsuarios.Click
    If frmUsuarios.Visible Then
        frmUsuarios.Activate()
    Else
        frmUsuarios.MdiParent = Me
        frmUsuarios.Show()
    End If
End Sub

Private Sub menuAreaLaboral_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles menuAreaLaboral.Click
    If frmAreasLaborales.Visible Then
        frmAreasLaborales.Activate()
    Else
        frmAreasLaborales.MdiParent = Me
        frmAreasLaborales.Show()
    End If
End Sub

Private Sub menuEmpleados_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles menuEmpleados.Click
    If frmEmpleados.Visible Then
        frmEmpleados.Activate()
    Else
        frmEmpleados.MdiParent = Me
        frmEmpleados.Show()
    End If
End Sub

Private Sub menuColores_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menuColores.Click
    If frmColores.Visible Then
        frmColores.Activate()
    Else
        frmColores.MdiParent = Me
        frmColores.Show()
    End If
End Sub

Private Sub menuMarcas_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menuMarcas.Click
    If frmMarcas.Visible Then
        frmMarcas.Activate()
    Else
        frmMarcas.MdiParent = Me
        frmMarcas.Show()
    End If
End Sub

```

```

Private Sub menuModelo_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menuModelo.Click
    If frmModelos.Visible Then
        frmModelos.Activate()
    Else
        frmModelos.MdiParent = Me
        frmModelos.Show()
    End If
End Sub

Private Sub menuListaAutos_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles menuListaAutos.Click
    If frmListadoAutos.Visible Then
        frmListadoAutos.Activate()
    Else
        frmListadoAutos.MdiParent = Me
        frmListadoAutos.Show()
    End If
End Sub

Private Sub menuClientes_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles menuClientes.Click
    If frmClientes.Visible Then
        frmClientes.Activate()
    Else
        frmClientes.MdiParent = Me
        frmClientes.Show()
    End If
End Sub

Private Sub menuListaEmpleados_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles menuListaEmpleados.Click
    If frmListadoEmpleados.Visible Then
        frmListadoEmpleados.Activate()
    Else
        frmListadoEmpleados.MdiParent = Me
        frmListadoEmpleados.Show()
    End If
End Sub

Private Sub menuProveedores_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles menuProveedores.Click
    If frmProveedores.Visible Then
        frmProveedores.Activate()
    Else
        frmProveedores.MdiParent = Me
        frmProveedores.Show()
    End If
End Sub

Private Sub menuListaCliente_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles menuListaCliente.Click
    If frmListadoClientes.Visible Then
        frmListadoClientes.Activate()
    Else

```



```

        frmListadoClientes.MdiParent = Me
        frmListadoClientes.Show()
    End If
End Sub

Private Sub menuListaProveedor_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles menuListaProveedor.Click
    If frmListadoProveedores.Visible Then
        frmListadoProveedores.Activate()
    Else
        frmListadoProveedores.MdiParent = Me
        frmListadoProveedores.Show()
    End If
End Sub

Private Sub menuRepuestos_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles menuRepuestos.Click
    If frmRepuestos.Visible Then
        frmRepuestos.Activate()
    Else
        frmRepuestos.MdiParent = Me
        frmRepuestos.Show()
    End If
End Sub

Private Sub menuServicios_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles menuServicios.Click
    If frmServicios.Visible Then
        frmServicios.Activate()
    Else
        frmServicios.MdiParent = Me
        frmServicios.Show()
    End If
End Sub

Private Sub menuPreciosProductos_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles menuPreciosProductos.Click
    If frmListaPrecioxProductos.Visible Then
        frmListaPrecioxProductos.Activate()
    Else
        frmListaPrecioxProductos.MdiParent = Me
        frmListaPrecioxProductos.producto = ""
        frmListaPrecioxProductos.Show()
    End If
End Sub

Private Sub menuListadoProductos_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles menuListadoProductos.Click
    If frmListadoProductos.Visible Then
        frmListadoProductos.Activate()
    Else
        frmListadoProductos.MdiParent = Me
        frmListadoProductos.Show()
    End If
End Sub

```

```

Private Sub menuGrupoProducto_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles menuGrupoProducto.Click
    If frmGruposProductos.Visible Then
        frmGruposProductos.Activate()
    Else
        frmGruposProductos.MdiParent = Me
        frmGruposProductos.Show()
    End If
End Sub

Private Sub menuCompras_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menuCompras.Click
    If frmExistencias.Visible Then
        frmExistencias.Activate()
    Else
        frmExistencias.MdiParent = Me
        frmExistencias.Show()
    End If
End Sub

Private Sub menuFormasPago_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs)
    If frmFormasPago.Visible Then
        frmFormasPago.Activate()
    Else
        frmFormasPago.MdiParent = Me
        frmFormasPago.Show()
    End If
End Sub

Private Sub menuProforma_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles menuProforma.Click
    If frmProforma.Visible Then
        frmProforma.Activate()
    Else
        frmProforma.MdiParent = Me
        frmProforma.Show()
    End If
End Sub

Private Sub menuAsesores_Click_1(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles menuAsesores.Click
    If frmAsesores.Visible Then
        frmAsesores.Activate()
    Else
        frmAsesores.MdiParent = Me
        frmAsesores.Show()
    End If
End Sub

Private Sub menuFormasPago_Click_1(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles menuFormasPago.Click
    If frmFormasPago.Visible Then
        frmFormasPago.Activate()

```

```

        Else
            frmFormasPago.MdiParent = Me
            frmFormasPago.Show()
        End If
    End Sub

    Private Sub menuAseguradoras_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles menuAseguradoras.Click
        If frmAseguradoras.Visible Then
            frmAseguradoras.Activate()
        Else
            frmAseguradoras.MdiParent = Me
            frmAseguradoras.Show()
        End If
    End Sub

    Private Sub menuOrdenTrabajo_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles menuOrdenTrabajo.Click
        If frmOrdenTrabajo.Visible Then
            frmOrdenTrabajo.Activate()
        Else
            frmOrdenTrabajo.MdiParent = Me
            frmOrdenTrabajo.Show()
        End If
    End Sub

    Private Sub menuFactura_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles menuFactura.Click
        If frmFacturacion.Visible Then
            frmFacturacion.Activate()
        Else
            frmFacturacion.MdiParent = Me
            frmFacturacion.Show()
        End If
    End Sub

    Private Sub menuRecibo_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
        If frmRecibo.Visible Then
            frmRecibo.Activate()
        Else
            frmRecibo.MdiParent = Me
            frmRecibo.Show()
        End If
    End Sub

    Private Sub menuRptOrdenesTrabajo_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles menuRptOrdenesTrabajo.Click
        If rptOrdenes_Trabajo.Visible Then
            rptOrdenes_Trabajo.Activate()
        Else
            rptOrdenes_Trabajo.MdiParent = Me
            rptOrdenes_Trabajo.Show()
        End If
    End Sub

```

```

Private Sub menuRptFacturas_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles menuRptFacturas.Click
    If rptFacturacion.Visible Then
        rptFacturacion.Activate()
    Else
        rptFacturacion.MdiParent = Me
        rptFacturacion.Show()
    End If
End Sub
End Class

```

Cambiar Contraseña

```

Public Class frmCambiarClave
    Private strSQL As String
    Private sql As New clsSQL

    Private Sub Limpiar()
        txtClave.Text = ""
        txClaveNueva.Text = ""
        txtConfirmacion.Text = ""
    End Sub

    Private Sub frmCambiarClave_Load(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        sql.Inicializar(ConnMaster)
        Limpiar()
    End Sub

    Private Sub tsbCancelar_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles tsbCancelar.Click
        Me.Close()
    End Sub

    Private Function ComprobarDatos() As Boolean
        Dim clave As String, nuevaclave As String, confirmaclave As String
        clave = txtClave.Text
        nuevaclave = txClaveNueva.Text
        confirmaclave = txtConfirmacion.Text

        If clave <> strClave Then
            MsgBox("La Contraseña Actual no coincide",
                MsgBoxStyle.Information, Me.Text)
            ComprobarDatos = False
        ElseIf nuevaclave = "" Then
            MsgBox("La Contraseña Nueva está vacía",
                MsgBoxStyle.Information, Me.Text)
            ComprobarDatos = False
        ElseIf confirmaclave <> nuevaclave Then
            MsgBox("La Confirmación de contraseña no coincide con la Contraseña Nueva",
                MsgBoxStyle.Information, Me.Text)
            ComprobarDatos = False
        Else
            ComprobarDatos = True
        End If
    End Function

```

End Function

```
Private Sub tsbAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles tsbAceptar.Click
    If ComprobarDatos() = True Then
        strSQL = " UPDATE usuario SET " & _
            " usu_clave='" & txClaveNueva.Text & "' " & _
            " WHERE usu_codigo='" & strUsuario & "' "
        sql.Ejecutar(strSQL)
        MsgBox("La contraseña fue cambiada correctamente",
MsgBoxStyle.Information, Me.Text)
        strClave = txClaveNueva.Text
    End If
End Sub
End Class
```

Usuarios

```
Public Class frmUsuarios
    Private strSQL As String
    Private SQL As New clsSQL
    Private cargado As Boolean
    Private cambiar As Boolean
    Dim ht As New Hashtable()
    Private Sub frmUsuarios_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        cargado = False
        Limpiar(True)
        cargado = True
    End Sub

    Private Sub btnCerrar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCerrar.Click
        Me.Close()
    End Sub

    Private Sub Limpiar(Optional ByVal todo As Boolean = False)
        If todo = False Then
            If tabControlUsuario.SelectedIndex = 0 Then
                txtNombre.Text = ""
                txtApellido.Text = ""
                txtUsuario.Text = ""
                txtClave.Text = ""
                txtCodigo.Text = ""
                txtClave.Enabled = True
                btnResetearClave.Enabled = False
                btnAceptarU.Text = "&Agregar"
                btnEliminarU.Enabled = False
                cambiar = False
                cargarUsuarios()
            Else
                cargarComboUsuarios()
                cargarGrupos()
            End If
        End If
    End Sub
End Class
```

```

        btnQuitaUno.Enabled = False
        btnAgregaUno.Enabled = False
        btnQuitaTodos.Enabled = False
        btnAgregaTodos.Enabled = False
    End If
Else
    txtNombre.Text = ""
    txtApellido.Text = ""
    txtUsuario.Text = ""
    txtClave.Text = ""
    txtCodigo.Text = ""
    txtClave.Enabled = True
    btnResetearClave.Enabled = False
    btnAceptarU.Text = "&Agregar"
    btnEliminarU.Enabled = False
    cambiar = False
    cargarUsuarios()
    cargarComboUsuarios()
    cargarGrupos()
    btnQuitaUno.Enabled = False
    btnAgregaUno.Enabled = False
    btnQuitaTodos.Enabled = False
    btnAgregaTodos.Enabled = False
End If
End Sub
Private Sub cargarUsuarios()
    Dim row As Integer, i As Integer
    strSQL = " SELECT usu_codigo,usu_nombre,usu_apellido" & _
            " FROM usuario " & _
            " ORDER BY usu_codigo,usu_nombre,usu_apellido "
    SQL.Ejecutar(strSQL)
    vsfg.Rows.Count = 1
    row = 1
    While SQL.dataR.Read()
        vsfg.Rows.Add()

        For i = 0 To SQL.dataR.FieldCount - 1
            vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
        Next
        row = row + 1
    End While
    SQL.dataR.Close()

    For i = 1 To vsfg.Rows.Count - 1
        vsfg.Item(i, 0) = CStr(i)
    Next
End Sub
Public Sub cargarComboUsuarios()
    strSQL = " SELECT usu_codigo as codigo,CONCAT(usu_codigo,' -
',TRIM(CONCAT(usu_apellido,' ',usu_nombre))) as nombre " & _
            " FROM usuario " & _
            " ORDER BY usu_apellido,usu_nombre "
    SQL.Ejecutar(strSQL)

    Dim ds As New DataTable
    ds.Load(SQL.dataR, LoadOption.OverwriteChanges)

```

```

cmbUsuario.DisplayMember = ds.Columns.Item(1).Caption
cmbUsuario.ValueMember = ds.Columns.Item(0).Caption
cmbUsuario.DataSource = ds

ds = Nothing
SQL.dataR.Close()
End Sub
Private Sub cargarGrupos()
    Dim usuario As String

    lstGrupos.Items.Clear()
    lstMiembro.Items.Clear()

    usuario = cmbUsuario.SelectedValue.ToString
    strSQL = " SELECT grupo.gru_codigo,gru_nombre " & _
            " FROM grupo " & _
            " ORDER BY gru_nombre "
    SQL.Ejecutar(strSQL)
    ht.Clear()
    While SQL.dataR.Read
        ht.Add(SQL.dataR.GetValue(1), SQL.dataR.GetValue(0))
        lstGrupos.Items.Add(SQL.dataR.GetValue(1))
    End While
    SQL.dataR.Close()

    strSQL = " SELECT grupo.gru_codigo,gru_nombre " & _
            " FROM grupo " & _
            " INNER JOIN grupo_usuario " & _
            " ON grupo.gru_codigo=grupo_usuario.gru_codigo " & _
            " INNER JOIN usuario " & _
            " ON grupo_usuario.usu_codigo=usuario.usu_codigo " & _
            " AND usuario.usu_codigo='" & usuario & "' " & _
            " ORDER BY grupo.gru_nombre "
    SQL.Ejecutar(strSQL)
    While SQL.dataR.Read
        lstMiembro.Items.Add(SQL.dataR.GetValue(1))
    End While
    SQL.dataR.Close()

    lstGrupos.SelectedIndex = -1
    lstMiembro.SelectedIndex = -1
    btnQuitaTodos.Enabled = False
    btnQuitaUno.Enabled = False
    btnAgregaUno.Enabled = False
    btnAgregaTodos.Enabled = False
End Sub

Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
    Limpiar()
End Sub
Private Sub vsfg_DoubleClick(ByVal sender As Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
    datosUsuario()
    txtClave.Enabled = False

```

```

    cambiar = False
    btnResetearClave.Enabled = True
    If txtUsuario.Text = "ADMIN" Or txtUsuario.Text = strUsuario Then
        btnEliminarU.Enabled = False
    Else
        btnEliminarU.Enabled = True
    End If
End Sub
Private Sub datosUsuario()
    Dim row As Integer
    row = vsfg.Row
    txtUsuario.Text = vsfg.Item(row, 1)
    txtNombre.Text = vsfg.Item(row, 2)
    txtApellido.Text = vsfg.Item(row, 3)
    txtClave.Text = vsfg.Item(row, 4)
    txtCodigo.Text = vsfg.Item(row, 1)
    btnAceptarU.Text = "&Modificar"
End Sub
Private Sub cmbGrupo_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
cmbUsuario.SelectedIndexChanged
    If cargado = True Then
        cargarGrupos()
    End If
End Sub

Private Sub btnAceptarU_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptarU.Click
    If btnAceptarU.Text = "&Agregar" Then
        If comprobarDatos("G") = True Then
            manipularDatos("I")
            MsgBox("Se ha ingresado el usuario correctamente",
MsgBoxStyle.Information, tabControlUsuario.SelectedTab.Text)
            Limpiar(True)
        End If
    Else
        If txtUsuario.Text = "" Then
            MsgBox("Seleccione el usuario que desea modificar",
MsgBoxStyle.Information, tabControlUsuario.SelectedTab.Text)
        Else
            manipularDatos("a")
            MsgBox("Se ha modificado el usuario correctamente",
MsgBoxStyle.Information, tabControlUsuario.SelectedTab.Text)
            Limpiar(True)
        End If
    End If
End Sub
Private Function comprobarDatos(ByVal tipo As String) As Boolean
    If tipo = "G" Then
        If Trim(txtUsuario.Text) = "" Then
            MsgBox("Ingrese un usuario válido",
MsgBoxStyle.Information, tabControlUsuario.SelectedTab.Text)
            Return False
        ElseIf Trim(txtNombre.Text) = "" Then

```



```

        MsgBox("Ingrese el nombre del usuario",
MsgBoxStyle.Information, tabControlUsuario.SelectedTab.Text)
        Return False
    ElseIf Trim(txtApellido.Text) = "" Then
        MsgBox("Ingrese el apellido del usuario",
MsgBoxStyle.Information, tabControlUsuario.SelectedTab.Text)
        Return False
    Else
        If btnAceptarU.Text = "&Agregar" Then
            For i As Integer = 1 To vsfg.Rows.Count - 1
                If Texto(txtUsuario.Text) = vsfg.Item(i, 1) Then
                    MsgBox("El usuario ya existe",
MsgBoxStyle.Information, tabControlUsuario.SelectedTab.Text)
                    Return False
                End If
            Next
            Return True
        Else
            Return True
        End If
    End If
Else
    If cmbUsuario.SelectedValue.ToString = "" Then
        MsgBox("Seleccione un usuario para asignarle grupos",
MsgBoxStyle.Information, tabControlUsuario.SelectedTab.Text)
        Return False
    Else
        Return True
    End If
End If
End Function
Private Sub manipularDatos(ByVal tipo As String)
    If UCase(tipo) = "I" Then
        strSQL = " INSERT INTO
usuario(usu_codigo,usu_nombre,usu_apellido,usu_clave) VALUES(" & _
        "" & Texto(txtUsuario.Text) & "','" & _
        Texto(txtNombre.Text) & "','" & Texto(txtApellido.Text) & "','" & _
        Texto(txtClave.Text) & "'"")
        SQL.Ejecutar(strSQL, "L", tipo)

    ElseIf UCase(tipo) = "E" Then
        strSQL = " DELETE FROM usuario " & _
        " WHERE usu_codigo='" & txtUsuario.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)

        strSQL = " DELETE FROM grupo_usuario " & _
        " WHERE usu_codigo='" & txtUsuario.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)

    Else
        strSQL = " UPDATE usuario SET " & _
        " usu_codigo='" & Texto(txtUsuario.Text) & "','" & _
        " usu_nombre='" & Texto(txtNombre.Text) & "','" & _
        " usu_apellido='" & Texto(txtApellido.Text) & "','" & _
        " usu_clave='" & Texto(txtClave.Text) & "' " & _
        " WHERE usu_codigo='" & Texto(txtCodigo.Text) & "' "

```

```

        SQL.Ejecutar(strSQL, "L", tipo)

    End If
End Sub

Private Sub btnEliminarU_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnEliminarU.Click
    If txtUsuario.Text = "" Then
        MsgBox("Seleccione el usuario que desea eliminar",
MsgBoxStyle.Information, tabControlUsuario.SelectedTab.Text)
    Else
        If MsgBox("Está seguro que desea eliminar el usuario " &
txtUsuario.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo,
tabControlUsuario.SelectedTab.Text) = MsgBoxResult.Yes Then
            manipularDatos("e")
            MsgBox("Se ha eliminado el usuario correctamente",
MsgBoxStyle.Information, tabControlUsuario.SelectedTab.Text)
            Limpiar(True)
        End If
    End If
End Sub

Private Sub btnAceptarP_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptarP.Click
    Dim i As Integer
    If comprobarDatos("P") = True Then
        strSQL = " DELETE FROM grupo_usuario " & _
            " WHERE usu_codigo = '" & cmbUsuario.SelectedValue &
            "' "
        SQL.Ejecutar(strSQL, "L", "E")

        For i = 0 To lstMiembro.Items.Count - 1

            strSQL = " INSERT INTO grupo_usuario
(usu_codigo,gru_codigo) " & _
                " VALUES('" & cmbUsuario.SelectedValue & "','" &
ht.Item(lstMiembro.Items.Item(i)) & "')"

            SQL.Ejecutar(strSQL, "L", "I")
        Next i
        MsgBox("Se han asignado los grupos al usuario correctamente",
MsgBoxStyle.Information, tabControlUsuario.SelectedTab.Text)
    End If
End Sub

Private Sub btnResetearClave_Click(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles btnResetearClave.Click
    If MsgBox("Está seguro que desea resetear la contraseña del
usuario?", MsgBoxStyle.Question + MsgBoxStyle.YesNo,
tabControlUsuario.SelectedTab.Text) = MsgBoxResult.Yes Then
        cambiar = True
        txtClave.Enabled = True
        txtClave.Focus()
    Else
        cambiar = False
    End If
End Sub

```

```

        txtClave.Enabled = False
    End If
End Sub

Private Sub btnAgregaUno_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnAgregaUno.Click
    Dim fila As Object, i As Integer, existe As Boolean
    fila = lstGrupos.SelectedItem
    existe = False

    For i = 0 To lstMiembro.Items.Count - 1
        If lstMiembro.Items.Item(i) = fila Then
            existe = True
            Exit For
        End If
    Next i
    If existe = False Then
        lstMiembro.Items.Add(fila)
    End If
End Sub

Private Sub btnAgregaTodos_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnAgregaTodos.Click
    Dim i As Integer, existe As Boolean
    Dim j As Integer
    Dim datos As Object

    For j = 0 To lstGrupos.Items.Count - 1
        existe = False
        For i = 0 To lstMiembro.Items.Count - 1
            If lstMiembro.Items.Item(i) = lstGrupos.Items.Item(j) Then
                existe = True
                Exit For
            End If
        Next i

        If existe = False Then
            datos = lstGrupos.Items.Item(j)
            lstMiembro.Items.Add(datos)
        End If
    Next j
End Sub

Private Sub btnQuitaUno_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnQuitaUno.Click
    If lstMiembro.SelectedIndex >= 0 Then
        lstMiembro.Items.RemoveAt(lstMiembro.SelectedIndex)
    End If

    If lstMiembro.Items.Count = 0 Then
        btnQuitaTodos.Enabled = False
        btnQuitaUno.Enabled = False
    End If
End Sub

```

```

Private Sub btnQuitaTodos_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnQuitaTodos.Click
    lstMiembro.Items.Clear()
    btnQuitaTodos.Enabled = False
    btnQuitaUno.Enabled = False
End Sub
Private Sub lstGrupos_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
lstGrupos.SelectedIndexChanged
    btnAgregaUno.Enabled = True
    btnAgregaTodos.Enabled = True
End Sub
Private Sub lstMiembro_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
lstMiembro.SelectedIndexChanged
    btnQuitaTodos.Enabled = True
    btnQuitaUno.Enabled = True
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
            .Cells(1, i).EntireRow.Font.Bold = True
            i += 1
        Next

        'i = 2

        Dim k As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            i = 1
            For row = 0 To vsfg.Rows.Count - 1
                .Cells(i, k).Value = vsfg.Item(row, col)
                i += 1
            Next
            k += 1
        Next
    End With
End Sub
End Class

```

Grupos

```
Imports System.Data.Odbc
```

```

Public Class frmGrupos
    Private strSQL As String
    Private SQL As New clsSQL
    Private cargado As Boolean

    Private Sub frmGrupos_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        cargado = False
        Limpiar(True)
        cargado = True
    End Sub
    Private Sub Limpiar(Optional ByVal todo As Boolean = False)
        If todo = False Then
            If tabControlGrupo.SelectedIndex = 0 Then
                txtDescripcion.Text = ""
                txtNombre.Text = ""
                txtCodigo.Text = ""
                btnAceptarG.Text = "&Agregar"
                btnEliminarG.Enabled = False
                cargarGrupos()
            Else
                cargarComboGrupos()
                cargarPermisos()
            End If
        Else
            txtDescripcion.Text = ""
            txtNombre.Text = ""
            txtCodigo.Text = ""
            btnAceptarG.Text = "&Agregar"
            btnEliminarG.Enabled = False
            cargarGrupos()
            cargarComboGrupos()
            cargarPermisos()
        End If
    End Sub
    Private Sub cargarGrupos()
        Dim row As Integer, i As Integer
        strSQL = " SELECT
gru_codigo,gru_nombre,COALESCE(gru_descripcion,') " & _
                " FROM grupo " & _
                " ORDER BY gru_nombre "
        SQL.Ejecutar(strSQL)
        vsfg.Rows.Count = 1
        row = 1
        While SQL.dataR.Read()
            vsfg.Rows.Add()

            For i = 0 To SQL.dataR.FieldCount - 1
                vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
            Next
            row = row + 1
        End While
        SQL.dataR.Close()

        For i = 1 To vsfg.Rows.Count - 1

```

```

        vsfg.Item(i, 0) = CStr(i)
    Next
End Sub
Public Sub cargarComboGrupos()
    strSQL = " SELECT gru_codigo as codigo,gru_nombre as nombre " & _
            " FROM grupo " & _
            " ORDER BY gru_nombre "
    SQL.Ejecutar(strSQL)
    Dim ds As New DataTable

    ds.Load(SQL.dataR, LoadOption.OverwriteChanges)

    cmbGrupo.DisplayMember = ds.Columns.Item(1).Caption
    cmbGrupo.ValueMember = ds.Columns.Item(0).Caption
    cmbGrupo.DataSource = ds

    ds = Nothing

    SQL.dataR.Close()
End Sub
Private Sub cargarPermisos()
    Dim row As Integer, i As Integer
    Dim grupo As String

    If cmbGrupo.Items.Count = 0 Then
        grupo = ""
    Else
        grupo = cmbGrupo.SelectedValue.ToString
    End If

    strSQL = " SELECT
menu.men_codigo,concat((space(length(menu.men_codigo)*3)),cast(menu.men_nom
bre as char(100))) as menu,IF(grupo_permiso.men_codigo is NULL,0,1) as
activo,men_obligatorio,IF(LENGTH(menu.men_codigo)=2,1,0) " & _
            " FROM menu " & _
            " LEFT JOIN grupo_permiso " & _
            " ON menu.men_codigo=grupo_permiso.men_codigo " & _
            " AND gru_codigo='" & grupo & "' " & _
            " WHERE men_plataforma='W' " & _
            " ORDER BY menu.men_codigo "
    SQL.Ejecutar(strSQL)
    vsfgPermisoWin.Rows.Count = 1
    row = 1
    While SQL.dataR.Read()
        vsfgPermisoWin.Rows.Add()

        For i = 0 To SQL.dataR.FieldCount - 1
            vsfgPermisoWin.Item(row, i) = Valor(SQL.dataR, i)
        Next
        row = row + 1
    End While
    SQL.dataR.Close()
    SQL.dataR.Close()

```

```

End Sub

Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
    Limpiar()
End Sub
Private Sub vsfg_DoubleClick(ByVal sender As Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
    datosGrupo()
    If txtCodigo.Text = "1" Then
        btnEliminarG.Enabled = False
    Else
        btnEliminarG.Enabled = True
    End If
End Sub
Private Sub datosGrupo()
    Dim row As Integer

    row = vsfg.Row
    txtCodigo.Text = vsfg.Item(row, 1)
    txtNombre.Text = vsfg.Item(row, 2)
    txtDescripcion.Text = vsfg.Item(row, 3)

    btnAceptarG.Text = "&Modificar"

End Sub

Private Sub cmbGrupo_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
cmbGrupo.SelectedIndexChanged
    If cargado = True Then
        cargarPermisos()
    End If
End Sub

Private Function comprobarDatos(ByVal tipo As String) As Boolean
    If tipo = "G" Then
        If Trim(txtNombre.Text) = "" Then
            MsgBox("Ingrese un nombre de grupo válido",
MsgBoxStyle.Information, tabControlGrupo.SelectedTab.Text)
            Return False
        Else
            Return True
        End If
    Else
        If cmbGrupo.SelectedValue.ToString = "" Then
            MsgBox("Seleccione un grupo para asignar permisos",
MsgBoxStyle.Information, tabControlGrupo.SelectedTab.Text)
            Return False
        Else
            Return True
        End If
    End If
End Function

Private Sub manipularDatos(ByVal tipo As String)

```

```

        If UCase(tipo) = "I" Then
            Dim cod As String = "1"
            strSQL = " INSERT INTO grupo(gru_nombre,gru_descripcion)
VALUES(" & _
                "' " & Texto(txtNombre.Text) & "','" &
                Texto(txtDescripcion.Text) & "'" )"

            SQL.Ejecutar(strSQL, "L", tipo)

            strSQL = " SELECT max(gru_codigo) FROM grupo "
            SQL.Ejecutar(strSQL)
            If SQL.dataR.HasRows Then
                SQL.dataR.Read()
                cod = Valor(SQL.dataR, 0)
            End If
            SQL.dataR.Close()

            strSQL = " INSERT INTO grupo_permiso(gru_codigo,men_codigo) " &
            _
                " SELECT " & cod & ",men_codigo" & _
                " FROM menu " & _
                " WHERE men_obligatorio=1 "
            SQL.Ejecutar(strSQL, "L", tipo)

        ElseIf UCase(tipo) = "E" Then
            strSQL = " DELETE FROM grupo " & _
                " WHERE gru_codigo='" & txtCodigo.Text & "' "
            SQL.Ejecutar(strSQL, "L", tipo)

            strSQL = " DELETE FROM grupo_permiso " & _
                " WHERE gru_codigo='" & txtCodigo.Text & "' "
            SQL.Ejecutar(strSQL, "L", tipo)

            strSQL = " DELETE FROM grupo_usuario " & _
                " WHERE gru_codigo='" & txtCodigo.Text & "' "
            SQL.Ejecutar(strSQL, "L", tipo)

        Else
            strSQL = " UPDATE grupo SET " & _
                " gru_nombre='" & Texto(txtNombre.Text) & "','" & _
                " gru_descripcion='" & Texto(txtDescripcion.Text) &
                "' " & _
                " WHERE gru_codigo='" & txtCodigo.Text & "' "
            SQL.Ejecutar(strSQL, "L", tipo)
        End If
    End Sub

    Private Sub btnAceptarP_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptarP.Click
        Dim i As Integer
        If comprobarDatos("P") = True Then
            strSQL = " DELETE FROM grupo_permiso " & _
                " WHERE gru_codigo = '" & cmbGrupo.SelectedValue & "'
            "
            SQL.Ejecutar(strSQL, "L", "E")
        End If
    End Sub

```



```

        For i = 1 To vsfgPermisoWin.Rows.Count - 1
            If CBool(vsfgPermisoWin.Item(i, 2)) Then
                strSQL = " INSERT INTO grupo_permiso
(gru_codigo,men_codigo) " & _
                    " VALUES('" & cmbGrupo.SelectedValue & "','" &
vsfgPermisoWin.Item(i, 0) & "'"
                SQL.Ejecutar(strSQL, "L", "I")

            End If
        Next i

        MsgBox("Se han asignado los permisos para el grupo",
MsgBoxStyle.Information, tabControlGrupo.SelectedTab.Text)
        Limpiar(True)
    End If
End Sub
Private Sub vsfgPermisoWin_AfterEdit(ByVal sender As Object, ByVal e As
Cl.Win.ClFlexGrid.RowColEventArgs) Handles vsfgPermisoWin.AfterEdit
    If e.Col = 2 Then
        Dim valor As Integer, Click As Integer = 0, i As Integer
        valor = Math.Abs(CInt(vsfgPermisoWin.Item(e.Row, 2)))
        If Math.Abs(CInt(vsfgPermisoWin.Item(e.Row, 4))) = 1 Then
            For i = e.Row + 1 To vsfgPermisoWin.Rows.Count - 1
                If Math.Abs(CInt(vsfgPermisoWin.Item(i, 4))) = 1 Then
Exit For
                    vsfgPermisoWin.Item(i, 2) = valor
                Next i
            ElseIf Math.Abs(CInt(vsfgPermisoWin.Item(e.Row, 4))) = 0 Then
                If valor = 1 Then
                    For i = e.Row To 1 Step -1
                        If Math.Abs(CInt(vsfgPermisoWin.Item(i, 4))) = 1
Then Click = i : Exit For
                            Next i
                        vsfgPermisoWin.Item(e.Row, 2) = valor
                        vsfgPermisoWin.Item(Click, 2) = valor
                    End If
                End If
            End If
        End Sub

        Private Sub vsfgPermisoWin_BeforeEdit(ByVal sender As System.Object,
ByVal e As Cl.Win.ClFlexGrid.RowColEventArgs) Handles
vsfgPermisoWin.BeforeEdit
            If e.Col <> 2 Then
                e.Cancel = True
            Else
                If Math.Abs(CInt(vsfgPermisoWin.Item(e.Row, 3))) = 1 Then
                    e.Cancel = True
                End If
            End If
        End Sub

        Private Sub btnCerrar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCerrar.Click
            Me.Close()

```

```

End Sub

Private Sub btnAceptarG_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptarG.Click
    If btnAceptarG.Text = "&Agregar" Then
        If comprobarDatos("G") = True Then
            manipularDatos("i")
            MsgBox("Se ha ingresado el grupo correctamente",
MsgBoxStyle.Information, tabControlGrupo.SelectedTab.Text)
            Limpiar(True)
        End If
    Else
        If txtCodigo.Text = "" Then
            MsgBox("Seleccione el grupo que desea modificar",
MsgBoxStyle.Information, tabControlGrupo.SelectedTab.Text)
        Else
            manipularDatos("a")
            MsgBox("Se ha modificado el grupo correctamente",
MsgBoxStyle.Information, tabControlGrupo.SelectedTab.Text)
            Limpiar(True)
        End If
    End If
End Sub

Private Sub btnEliminarG_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles btnEliminarG.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione el grupo que desea eliminar",
MsgBoxStyle.Information, tabControlGrupo.SelectedTab.Text)
    Else
        If MsgBox("Está seguro que desea eliminar el grupo " &
txtNombre.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo,
tabControlGrupo.SelectedTab.Text) = MsgBoxResult.Yes Then
            manipularDatos("e")
            MsgBox("Se ha eliminado el grupo correctamente",
MsgBoxStyle.Information, tabControlGrupo.SelectedTab.Text)
            Limpiar(True)
        End If
    End If
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
            .Cells(1, i).EntireRow.Font.Bold = True
        Next
    End With
End Sub

```

```

        i += 1
    Next

    'i = 2

    Dim k As Integer = 1
    For col = 0 To vsfg.Cols.Count - 1
        i = 1
        For row = 0 To vsfg.Rows.Count - 1
            .Cells(i, k).Value = vsfg.Item(row, col)
            i += 1
        Next
        k += 1
    Next

    End With
End Sub
End Class

```

Areas Laborales

```

Public Class frmAreasLaborales
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "el Área Laboral"
    Dim strExcel As String
    Private Sub frmAreasLaborales_Load(ByVal sender As System.Object, ByVal
e As System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        Limpiar()
    End Sub
    Private Sub Limpiar()
        txtCodigo.Text = ""
        txtNombre.Text = ""
        txtDescripcion.Text = ""
        btnAceptar.Text = "&Aceptar"
        txtNombre.Focus()
        btnEliminar.Enabled = False
        Cargar()
    End Sub
    Private Sub Cargar()
        Dim row As Integer, i As Integer
        strSQL = " SELECT are_lab_codigo,are_lab_nombre,are_lab_descripcion
" & _
                " FROM area_laboral " & _
                " ORDER BY are_lab_nombre "

        SQL.Ejecutar(strSQL)
        vsfg.Rows.Count = 1
        row = 1
        While SQL.dataR.Read()
            vsfg.Rows.Add()
            For i = 0 To SQL.dataR.FieldCount - 1
                vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
            Next
        End While
    End Sub
End Class

```

```

        Next
        row = row + 1
    End While
    SQL.dataR.Close()

    For i = 1 To vsfg.Rows.Count - 1
        vsfg.Item(i, 0) = CStr(i)
    Next
End Sub
Private Sub datosColor()
    Dim row As Integer
    row = vsfg.Row

    txtCodigo.Text = vsfg.Item(row, 1)
    txtNombre.Text = vsfg.Item(row, 2)
    txtDescripcion.Text = vsfg.Item(row, 3)

    btnAceptar.Text = "&Modificar"
    txtCodigo.Enabled = False
    btnEliminar.Enabled = True
End Sub

Private Sub vsfg_DoubleClick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
    datosColor()
End Sub
Private Function controlDatos() As Boolean
    If Trim(txtNombre.Text) = "" Then
        MsgBox("Ingrese un nombre para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtNombre.Focus()
        Return False
    Else
        Return True
    End If
End Function

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If btnAceptar.Text = "&Aceptar" Then
        If controlDatos() = True Then
            manipularDatos("i")
            MsgBox("Se han ingresado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    Else
        If controlDatos() = True Then
            manipularDatos("a")
            MsgBox("Se han modificado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    End If
End Sub

```

```

Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione " & Ventana & " que desea eliminar",
MsgBoxStyle.Information, Me.Text)
    Else
        Dim Cuantos As Integer = 0
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
            " FROM empleado " & _
            " WHERE are_lab_codigo='" & txtCodigo.Text & "'"

        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
        SQL.dataR.Close()
        If Cuantos > 0 Then
            MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionada con datos de Empleados, no se
puede eliminar", MsgBoxStyle.Information, Me.Text)
            Exit Sub
        End If

        If MsgBox("Está seguro que desea eliminar " & Ventana & " " &
txtNombre.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
            manipularDatos("e")
            MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    End If
End Sub

Private Sub manipularDatos(ByVal tipo As String)
    If UCase(tipo) = "I" Then
        strSQL = " INSERT INTO
area_laboral(are_lab_nombre,are_lab_descripcion) VALUES('" & _
            Texto(txtNombre.Text) & "','" & _
            Texto(txtDescripcion.Text) & "'" )
        SQL.Ejecutar(strSQL, "L", tipo)

    ElseIf UCase(tipo) = "E" Then
        strSQL = " DELETE FROM area_laboral " & _
            " WHERE are_lab_codigo='" & txtCodigo.Text & "'"
        SQL.Ejecutar(strSQL, "L", tipo)

    Else
        strSQL = " UPDATE area_laboral SET " & _
            " are_lab_nombre='" & Texto(txtNombre.Text) & "','" & _
            " are_lab_descripcion='" & Texto(txtDescripcion.Text)
& "'" & _
            " WHERE are_lab_codigo='" & txtCodigo.Text & "'"
        SQL.Ejecutar(strSQL, "L", tipo)
    End If
End Sub

```

```

        End If
    End Sub

    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
        Limpiar()
    End Sub

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

    Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
        Dim excel As New Excel.Application
        excel.Visible = True

        With excel
            .SheetsInNewWorkbook = 1
            .Workbooks.Add()
            .Worksheets(1).Select()

            Dim i As Integer = 1
            For col = 0 To vsfg.Cols.Count - 1
                .Cells(1, i).value = vsfg.Item(0, col)
                .Cells(1, i).EntireRow.Font.Bold = True
                i += 1
            Next

            'i = 2

            Dim k As Integer = 1
            For col = 0 To vsfg.Cols.Count - 1
                i = 1
                For row = 0 To vsfg.Rows.Count - 1
                    .Cells(i, k).Value = vsfg.Item(row, col)
                    i += 1
                Next
                k += 1
            Next
        End With
    End Sub
End Class

```

Empleados

```

Public Class frmEmpleados
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "el Empleado"
    Private Sub frmEmpleados_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        cargarCombos()
    End Sub
End Class

```

```

        Limpiar()
    End Sub
    Public Sub cargarCombos()
        Dim ds0 As New DataTable
        strSQL = " SELECT 1,'0' as codigo,'Seleccione una...' as nombre " &
        " UNION " & _
        " SELECT 2,cast(are_lab_codigo as char) as codigo,are_lab_nombre
as nombre " & _
        " FROM area_laboral "
        SQL.Ejecutar(strSQL)
        ds0.Load(SQL.dataR, LoadOption.OverwriteChanges)
        comboAreaLaboral.DisplayMember = ds0.Columns.Item(2).Caption
        comboAreaLaboral.ValueMember = ds0.Columns.Item(1).Caption
        comboAreaLaboral.DataSource = ds0
        SQL.dataR.Close()
    End Sub

    Private Sub Limpiar()
        txtNombre.Text = ""
        txtApellido.Text = ""
        txtRuc.Text = ""
        txtDireccion.Text = ""
        txtTelefono.Text = ""
        comboAreaLaboral.SelectedIndex = 0
        txtCelular.Text = ""
        txtEMail.Text = ""
        txtCodigo.Text = ""
        txtActualiza.Text = ""
        btnAceptar.Text = "&Aceptar"
        btnEliminar.Enabled = False
        Cargar()
        txtNombre.Focus()
    End Sub
    Private Sub Cargar()
        Dim row As Integer, i As Integer
        strSQL = " SELECT
epl_codigo,epl_nombre,epl_apellido,empleado.are_lab_codigo,are_lab_nombre,e
pl_ruc,epl_direccion,epl_telefono,epl_celular,epl_email" & _
        " FROM empleado " & _
        " LEFT JOIN area_laboral " & _
        " ON area_laboral.are_lab_codigo=empleado.are_lab_codigo " &
        " ORDER BY epl_apellido,epl_nombre "
        SQL.Ejecutar(strSQL)
        vsfg.Rows.Count = 1
        row = 1
        While SQL.dataR.Read()
            vsfg.Rows.Add()
            For i = 0 To SQL.dataR.FieldCount - 1
                vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
            Next
            row = row + 1
        End While
        SQL.dataR.Close()
    End Sub

```

```

        For i = 1 To vsfg.Rows.Count - 1
            vsfg.Item(i, 0) = CStr(i)
        Next
    End Sub

    Private Function controlDatos() As Boolean
        If Trim(txtNombre.Text) = "" Then
            MsgBox("Ingreso el nombre para " & Ventana,
                MsgBoxStyle.Information, Me.Text)
            txtNombre.Focus()
            Return False
        ElseIf Trim(txtApellido.Text) = "" Then
            MsgBox("Ingreso el apellido para " & Ventana,
                MsgBoxStyle.Information, Me.Text)
            txtApellido.Focus()
            Return False
        Else
            Return True
        End If
    End Function

    Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
        System.EventArgs) Handles btnAceptar.Click
        txtIngresa.Text = "i"
        txtActualiza.Text = "a"

        If btnAceptar.Text = "&Aceptar" Then
            If controlDatos() = True Then
                If VerificarID(txtRuc.Text) Then
                    manipularDatos("i")
                    MsgBox("Se han ingresado los datos correctamente",
                        MsgBoxStyle.Information, Me.Text)
                    Limpiar()
                    'End If
                End If
            End If
        Else
            If controlDatos() = True Then
                If VerificarID(txtRuc.Text) Then
                    SQL.dataR.Close()
                    manipularDatos("a")
                    MsgBox("Se han modificado los datos correctamente",
                        MsgBoxStyle.Information, Me.Text)
                    Limpiar()
                End If
            End If
        End If
    End Sub

    Private Sub manipularDatos(ByVal tipo As String)
        If UCase(tipo) = "I" Then
            strSQL = "INSERT INTO
empleado(epl_nombre,epl_apellido,are_lab_codigo,epl_ruc," & _
"epl_direccion,epl_telefono,epl_celular,epl_email) VALUES(' " & _
                Texto(txtNombre.Text) & "','" & _

```



```

        Texto(txtApellido.Text) & "','" & _
        comboAreaLaboral.SelectedIndex & "','" & _
        Trim(txtRuc.Text) & "','" & _
        Texto(txtDireccion.Text) & "','" & _
        Trim(txtTelefono.Text) & "','" & _
        Trim(txtCelular.Text) & "','" & _
        Trim(txtEMail.Text) & "'"")
    SQL.Ejecutar(strSQL, "L", tipo)

    ElseIf UCase(tipo) = "E" Then
        strSQL = " DELETE FROM empleado " & _
            " WHERE epl_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)

    Else
        strSQL = " UPDATE empleado SET " & _
            " epl_nombre='" & Texto(txtNombre.Text) & "','" & _
            " epl_apellido='" & Texto(txtApellido.Text) & "','" & _
            " are_lab_codigo='" & comboAreaLaboral.SelectedIndex &
            "','" & _
            " epl_ruc='" & Trim(txtRuc.Text) & "','" & _
            " epl_direccion='" & Texto(txtDireccion.Text) & "','" & _
            " epl_telefono='" & Trim(txtTelefono.Text) & "','" & _
            " epl_celular='" & Trim(txtCelular.Text) & "','" & _
            " epl_email='" & Trim(txtEMail.Text) & "'" & _
            " WHERE epl_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)
    End If
End Sub

Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub

Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
    Limpiar()
End Sub

Private Sub vsfg_DoubleClick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
    datosAsesor()
End Sub

Private Sub datosAsesor()
    Dim row As Integer
    row = vsfg.Row

    txtCodigo.Text = vsfg.Item(row, 1)
    txtNombre.Text = vsfg.Item(row, 2)
    txtApellido.Text = vsfg.Item(row, 3)
    comboAreaLaboral.SelectedValue = vsfg.Item(row, 4)
    txtRuc.Text = vsfg.Item(row, 6)
    txtDireccion.Text = vsfg.Item(row, 7)
    txtTelefono.Text = vsfg.Item(row, 8)
    txtCelular.Text = vsfg.Item(row, 9)
    txtEMail.Text = vsfg.Item(row, 10)

```

```

        btnAceptar.Text = "&Modificar"
        btnEliminar.Enabled = True
    End Sub

    Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
        If txtCodigo.Text = "" Then
            MsgBox("Seleccione " & Ventana & " que desea eliminar",
MsgBoxStyle.Information, Me.Text)
        Else
            If MsgBox("Está seguro que desea eliminar " & Ventana & " " &
txtApellido.Text & " " & txtNombre.Text & "?", MsgBoxStyle.Question +
MsgBoxStyle.YesNo, Me.Text) = MsgBoxResult.Yes Then
                manipularDatos("e")
                MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)
                Limpiar()
            End If
        End If
    End Sub

    Private Sub txtTelefono_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtTelefono.KeyPress
        If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
            e.KeyChar = ""
        End If
    End Sub

    Private Sub txtCelular_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtCelular.KeyPress
        If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
            e.KeyChar = ""
        End If
    End Sub

    Private Sub txtRuc_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtRuc.KeyPress
        If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
            e.KeyChar = ""
        End If
    End Sub

    Private Sub txtRuc_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtRuc.LostFocus

        If (txtActualiza.Text <> "a" Or txtIngresa.Text <> "i") Then
            Dim Cuantos As Integer = 0

            strSQL = " SELECT COALESCE(COUNT(*),0) " & _
                " FROM empleado " & _
                " WHERE ep1_ruc='" & Trim(txtRuc.Text) & "' "
            SQL.Ejecutar(strSQL)
            If SQL.dataR.HasRows Then
                Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
            End If
            SQL.dataR.Close()
            If Cuantos > 0 Then

```

```

        If MsgBox("Este CI/RUC ya está registrado, desea cargar la
información actual?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
            CargarRuc(Trim(txtRuc.Text))
        Else
            txtRuc.Text = ""
            txtRuc.Focus()
        End If
    End If
End Sub
Private Sub CargarRuc(ByVal txt1 As String)
    Dim row As Integer
    row = vsfg.Row
    Limpiar()

    strSQL = "SELECT
epl_codigo,epl_nombre,epl_apellido,epl_ruc,epl_direccion,epl_telefono,epl_c
elular,epl_email,are_lab_codigo" & _
" FROM empleado " & _
"WHERE epl_ruc='" & txt1 & "'"

    SQL.Ejecutar(strSQL)
    If SQL.dataR.HasRows Then
        SQL.dataR.Read()
        txtNombre.Text = SQL.dataR.GetValue(1)
        txtApellido.Text = Valor(SQL.dataR, 2)
        txtRuc.Text = SQL.dataR.GetValue(3)
        txtDireccion.Text = Valor(SQL.dataR, 4)
        txtTelefono.Text = Valor(SQL.dataR, 5)
        txtCelular.Text = Valor(SQL.dataR, 6)
        txtEMail.Text = Valor(SQL.dataR, 7)
        comboAreaLaboral.SelectedValue = Valor(SQL.dataR, 8)

        btnAceptar.Text = "&Modificar"
        btnEliminar.Enabled = True
        txtCodigo.Text = SQL.dataR.GetValue(0)
        SQL.dataR.Close()
    End If
    SQL.dataR.Close()

End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1

```

```

        .Cells(1, i).value = vsfg.Item(0, col)
        .Cells(1, i).EntireRow.Font.Bold = True
        i += 1
    Next

    Dim k As Integer = 1
    For col = 0 To vsfg.Cols.Count - 1
        i = 1
        For row = 0 To vsfg.Rows.Count - 1
            .Cells(i, k).Value = vsfg.Item(row, col)
            i += 1
        Next
        k += 1
    Next
End With
End Sub
End Class

```

Listado Empleados

```

Public Class frmListadoEmpleados
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "el Empleado"
    Private Sub frmListadoEmpleados_Load(ByVal sender As System.Object,
    ByVal e As System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        cargarCombos()
        Limpiar()
    End Sub
    Public Sub cargarCombos()
        Dim ds0 As New DataTable
        strSQL = " SELECT 1,'% ' as codigo,'Todas las Áreas' as nombre " & _
            " UNION " & _
            " SELECT 2,cast(are_lab_codigo as char) as codigo,are_lab_nombre
as nombre " & _
            " FROM area_laboral " & _
            " ORDER BY 1,nombre "
        SQL.Ejecutar(strSQL)
        ds0.Load(SQL.dataR, LoadOption.OverwriteChanges)
        comboAreaLaboral.DisplayMember = ds0.Columns.Item(2).Caption
        comboAreaLaboral.ValueMember = ds0.Columns.Item(1).Caption
        comboAreaLaboral.DataSource = ds0
        SQL.dataR.Close()
    End Sub

    Private Sub Limpiar()
        txtRucFiltro.Text = ""
        txtNombreFiltro.Text = ""
        comboAreaLaboral.SelectedIndex = 0
        Cargar()
        vsfg.Focus()
    End Sub

```

```

Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub

Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
    Limpiar()
End Sub

Private Sub Cargar()
    Dim row As Integer, i As Integer
    strSQL = " SELECT
epl_codigo,epl_nombre,epl_apellido,are_lab_nombre,epl_ruc,epl_direccion,epl
_telefono,epl_celular,epl_email " & _
" FROM empleado " & _
" LEFT JOIN area_laboral " & _
" ON area_laboral.are_lab_codigo=empleado.are_lab_codigo " & _
" WHERE (epl_nombre LIKE '%" & txtNombreFiltro.Text & '%" OR
epl_apellido LIKE '%" & txtNombreFiltro.Text & '%" ) " & _
" AND epl_ruc LIKE '%" & txtRucFiltro.Text & '%" " & _
" AND empleado.are_lab_codigo LIKE '" &
comboAreaLaboral.SelectedValue.ToString & "' " & _
" ORDER BY epl_apellido,epl_nombre "
    SQL.Ejecutar(strSQL)
    vsfg.Rows.Count = 1
    row = 1
    While SQL.dataR.Read()
        vsfg.Rows.Add()
        For i = 0 To SQL.dataR.FieldCount - 1
            vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
        Next
        row = row + 1
    End While
    SQL.dataR.Close()

    For i = 1 To vsfg.Rows.Count - 1
        vsfg.Item(i, 0) = CStr(i)
    Next
    'End If

End Sub

Private Sub btnBuscar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnBuscar.Click
    If txtRucFiltro.Text <> "" Then
        If VerificarID(txtRucFiltro.Text) Then
            Cargar()
        End If
    Else
        Cargar()
    End If
End Sub

Private Sub txtRucFiltro_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtRucFiltro.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then

```

```

        e.KeyChar = ""
    End If
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
            .Cells(1, i).EntireRow.Font.Bold = True
            i += 1
        Next

        Dim k As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            i = 1
            For row = 0 To vsfg.Rows.Count - 1
                .Cells(i, k).Value = vsfg.Item(row, col)
                i += 1
            Next
            k += 1
        Next

    End With
End Sub
End Class

```

Colores

```

Public Class frmColores
    Private strSQL As String
    Private SQL As New clsSQL
    Private Sub frmColores_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        Limpiar()
    End Sub
    Private Sub Limpiar()
        txtCodigo.Text = ""
        txtNombre.Text = ""
        txtDescripcion.Text = ""
        btnAceptar.Text = "&Aceptar"
        txtNombre.Focus()
        btnEliminar.Enabled = False
        Cargar()
    End Sub
    Private Sub Cargar()

```

```

Dim row As Integer, i As Integer
strSQL = " SELECT col_codigo,col_nombre,col_descripcion " & _
         " FROM color " & _
         " ORDER BY col_nombre "

SQL.Ejecutar(strSQL)
vsfg.Rows.Count = 1
row = 1
While SQL.dataR.Read()
    vsfg.Rows.Add()
    For i = 0 To SQL.dataR.FieldCount - 1
        vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
    Next
    row = row + 1
End While
SQL.dataR.Close()

For i = 1 To vsfg.Rows.Count - 1
    vsfg.Item(i, 0) = CStr(i)
Next
End Sub
Private Sub datosColor()
    Dim row As Integer
    row = vsfg.Row

    txtCodigo.Text = vsfg.Item(row, 1)
    txtNombre.Text = vsfg.Item(row, 2)
    txtDescripcion.Text = vsfg.Item(row, 3)

    btnAceptar.Text = "&Modificar"
    txtCodigo.Enabled = False
    btnEliminar.Enabled = True
End Sub

Private Sub vsfg_DoubleClick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
    datosColor()
End Sub
Private Function controlDatos() As Boolean
    If Trim(txtNombre.Text) = "" Then
        MsgBox("Ingrese un nombre para el Color",
MsgBoxStyle.Information, Me.Text)
        txtNombre.Focus()
        Return False
    Else
        Return True
    End If
End Function

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If btnAceptar.Text = "&Aceptar" Then
        If controlDatos() = True Then
            manipularDatos("i")
            MsgBox("Se han ingresado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
        End If
    End If
End Sub

```

```

        Limpiar()
    End If
Else
    If controlDatos() = True Then
        manipularDatos("a")
        MsgBox("Se han modificado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
        Limpiar()
    End If
End If
End Sub
Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione el Color que desea eliminar",
MsgBoxStyle.Information, Me.Text)
    Else
        Dim Cuantos As Integer
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
                " FROM auto " & _
                " WHERE col_codigo='" & txtCodigo.Text & "'"

        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
        SQL.dataR.Close()
        If Cuantos > 0 Then
            If MsgBox("El Color está relacionado con datos de Autos,
desea continuar?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.No Then
                Exit Sub
            Else
                manipularDatos("e")
                MsgBox("Se ha eliminado el color correctamente",
MsgBoxStyle.Information, Me.Text)
                Limpiar()
            End If
        Else
            If MsgBox("Está seguro que desea eliminar el Color " &
txtNombre.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
                manipularDatos("e")
                MsgBox("Se ha eliminado el color correctamente",
MsgBoxStyle.Information, Me.Text)
                Limpiar()
            End If
        End If
    End If
End Sub

Private Sub manipularDatos(ByVal tipo As String)
    If UCase(tipo) = "I" Then
        strSQL = " INSERT INTO color(col_nombre,col_descripcion)
VALUES(' " & _
                Texto(txtNombre.Text) & "',' " & _
                Texto(txtDescripcion.Text) & "'"

```



```

        SQL.Ejecutar(strSQL, "L", tipo)

    ElseIf UCase(tipo) = "E" Then
        strSQL = " DELETE FROM color " & _
            " WHERE col_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)
        strSQL = " DELETE FROM auto " & _
            " WHERE col_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)
    Else
        strSQL = " UPDATE color SET " & _
            " col_nombre='" & Texto(txtNombre.Text) & "'," & _
            " col_descripcion='" & Texto(txtDescripcion.Text) & "' "
& _
            " WHERE col_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)
    End If
End Sub

Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub

Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
    Limpiar()
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
            .Cells(1, i).EntireRow.Font.Bold = True
            i += 1
        Next

        Dim k As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            i = 1
            For row = 0 To vsfg.Rows.Count - 1
                .Cells(i, k).Value = vsfg.Item(row, col)
                i += 1
            Next
            k += 1
        Next
    End With
End Sub

```

End Class

Marcas

```
Public Class frmMarcas
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "la Marca"
    Private Sub frmMarcas_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        Limpiar()
    End Sub

    Private Sub Limpiar()
        txtCodigo.Text = ""
        txtNombre.Text = ""
        txtDescripcion.Text = ""
        btnAceptar.Text = "&Aceptar"
        txtNombre.Focus()
        btnEliminar.Enabled = False
        cargar()
    End Sub

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
        Limpiar()
    End Sub

    Private Sub Cargar()
        Dim row As Integer, i As Integer
        strSQL = " SELECT mar_codigo,mar_nombre,mar_descripcion " & _
            " FROM marca " & _
            " ORDER BY mar_nombre "

        SQL.Ejecutar(strSQL)
        vsfg.Rows.Count = 1
        row = 1
        While SQL.dataR.Read()
            vsfg.Rows.Add()
            For i = 0 To SQL.dataR.FieldCount - 1
                vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
            Next
            row = row + 1
        End While
        SQL.dataR.Close()

        For i = 1 To vsfg.Rows.Count - 1
            vsfg.Item(i, 0) = CStr(i)
        Next
    End Sub

    Private Sub datosColor()
```

```

Dim row As Integer
row = vsfg.Row

txtCodigo.Text = vsfg.Item(row, 1)
txtNombre.Text = vsfg.Item(row, 2)
txtDescripcion.Text = vsfg.Item(row, 3)

btnAceptar.Text = "&Modificar"
txtCodigo.Enabled = False
btnEliminar.Enabled = True
End Sub

Private Sub vsfg_DoubleClick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
    datosColor()
End Sub
Private Function controlDatos() As Boolean
    If Trim(txtNombre.Text) = "" Then
        MsgBox("Ingrese un nombre para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtNombre.Focus()
        Return False
    Else
        Return True
    End If
End Function

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If btnAceptar.Text = "&Aceptar" Then
        If controlDatos() = True Then
            manipularDatos("i")
            MsgBox("Se han ingresado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    Else
        If controlDatos() = True Then
            manipularDatos("a")
            MsgBox("Se han modificado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    End If
End Sub
Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione " & Ventana & " que desea eliminar",
MsgBoxStyle.Information, Me.Text)
    Else
        Dim Cuantos As Integer = 0
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
            " FROM modelo " & _
            " WHERE mar_codigo='" & txtCodigo.Text & "'"
        SQL.Ejecutar(strSQL)
    End If
End Sub

```

```

        SQL.dataR.Read()
        Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
        SQL.dataR.Close()
        If Cuantos > 0 Then
            MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionada con datos de Modelos, no se puede
eliminar", MsgBoxStyle.Information, Me.Text)
            Exit Sub
        End If

        If MsgBox("Está seguro que desea eliminar " & Ventana & " " &
txtNombre.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
            manipularDatos("e")
            MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    End If
End Sub

Private Sub manipularDatos(ByVal tipo As String)
    If UCase(tipo) = "I" Then
        strSQL = " INSERT INTO marca(mar_nombre,mar_descripcion)
VALUES(' " & _
                Texto(txtNombre.Text) & "','" & _
                Texto(txtDescripcion.Text) & "')"
        SQL.Ejecutar(strSQL, "L", tipo)

    ElseIf UCase(tipo) = "E" Then
        strSQL = " DELETE FROM marca " & _
                " WHERE mar_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)

    Else
        strSQL = " UPDATE marca SET " & _
                " mar_nombre='" & Texto(txtNombre.Text) & "','" & _
                " mar_descripcion='" & Texto(txtDescripcion.Text) &
                "' " & _
                " WHERE mar_codigo='" & txtCodigo.Text & "' "

        SQL.Ejecutar(strSQL, "L", tipo)
    End If
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()
    End With
End Sub

```

```

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
            .Cells(1, i).EntireRow.Font.Bold = True
            i += 1
        Next

        Dim k As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            i = 1
            For row = 0 To vsfg.Rows.Count - 1
                .Cells(i, k).Value = vsfg.Item(row, col)
                i += 1
            Next
            k += 1
        Next
    End With
End Sub
End Class

```

Modelos

```

Public Class frmModelos
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "el Modelo"
    Private Sub frmModelos_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        cargarCombo()
        Limpiar()
    End Sub
    Public Sub cargarCombo()
        strSQL = " SELECT mar_codigo as codigo,mar_nombre as nombre " & _
            " FROM marca " & _
            " ORDER BY mar_nombre "
        SQL.Ejecutar(strSQL)

        Dim ds As New DataTable
        ds.Load(SQL.dataR, LoadOption.OverwriteChanges)
        combo.DisplayMember = ds.Columns.Item(1).Caption
        combo.ValueMember = ds.Columns.Item(0).Caption
        combo.DataSource = ds

        ds = Nothing
        SQL.dataR.Close()
    End Sub

    Private Sub Limpiar()
        txtCodigo.Text = ""
        txtNombre.Text = ""
        txtDescripcion.Text = ""
        btnAceptar.Text = "&Aceptar"
        txtNombre.Focus()
        btnEliminar.Enabled = False
    End Sub

```

```

        Cargar()
    End Sub

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
        Limpiar()
    End Sub
    Private Sub Cargar()
        Dim row As Integer, i As Integer
        strSQL = " SELECT mod_codigo,mod_nombre,mod_descripcion " & _
                " FROM modelo " & _
                " WHERE mar_codigo='" & combo.SelectedValue.ToString & "'
" & _
                " ORDER BY mod_nombre "

        SQL.Ejecutar(strSQL)
        vsfg.Rows.Count = 1
        row = 1
        While SQL.dataR.Read()
            vsfg.Rows.Add()
            For i = 0 To SQL.dataR.FieldCount - 1
                vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
            Next
            row = row + 1
        End While
        SQL.dataR.Close()

        For i = 1 To vsfg.Rows.Count - 1
            vsfg.Item(i, 0) = CStr(i)
        Next
    End Sub

    Private Sub datosColor()
        Dim row As Integer
        row = vsfg.Row

        txtCodigo.Text = vsfg.Item(row, 1)
        txtNombre.Text = vsfg.Item(row, 2)
        txtDescripcion.Text = vsfg.Item(row, 3)

        btnAceptar.Text = "&Modificar"
        txtCodigo.Enabled = False
        btnEliminar.Enabled = True
    End Sub

    Private Sub vsfg_DoubleClick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
        datosColor()
    End Sub
    Private Function controlDatos() As Boolean
        If Trim(txtNombre.Text) = "" Then

```

```

        MsgBox("Ingrese un nombre para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtNombre.Focus()
        Return False
    Else
        Return True
    End If
End Function

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If btnAceptar.Text = "&Aceptar" Then
        If controlDatos() = True Then
            manipularDatos("i")
            MsgBox("Se han ingresado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    Else
        If controlDatos() = True Then
            manipularDatos("a")
            MsgBox("Se han modificado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    End If
End Sub

Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione " & Ventana & " que desea eliminar",
MsgBoxStyle.Information, Me.Text)
    Else
        Dim Cuantos As Integer
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
                " FROM auto " & _
                " WHERE mod_codigo='" & txtCodigo.Text & "'"

        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
        SQL.dataR.Close()
        If Cuantos > 0 Then
            MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de Autos, no se puede
eliminar", MsgBoxStyle.Information, Me.Text)
            Exit Sub
        End If

        If MsgBox("Está seguro que desea eliminar " & Ventana & " " &
txtNombre.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
            manipularDatos("e")
            MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)

```

```

        Limpiar()
    End If
End If
End Sub

Private Sub manipularDatos(ByVal tipo As String)
    If UCase(tipo) = "I" Then
        Dim Cuantos As Integer = 1
        strSQL = " SELECT COALESCE(MAX(mod_codigo)+1,1) " & _
            " FROM modelo "

        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        If SQL.dataR.HasRows Then
            Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
        End If
        SQL.dataR.Close()
        strSQL = " INSERT INTO
modelo(mar_codigo,mod_codigo,mod_nombre,mod_descripcion) VALUES(' " & _
            combo.SelectedValue & "','" & _
            Cuantos.ToString & "','" & _
            Texto(txtNombre.Text) & "','" & _
            Texto(txtDescripcion.Text) & "'" )"
        SQL.Ejecutar(strSQL, "L", tipo)

    ElseIf UCase(tipo) = "E" Then
        strSQL = " DELETE FROM modelo " & _
            " WHERE mod_codigo='" & txtCodigo.Text & "' " & _
            " AND mar_codigo='" & combo.SelectedValue & "' "
        SQL.Ejecutar(strSQL, "L", tipo)

    Else
        strSQL = " UPDATE modelo SET " & _
            " mod_nombre='" & Texto(txtNombre.Text) & "','" & _
            " mod_descripcion='" & Texto(txtDescripcion.Text) & _
            "'" & _
            " WHERE mod_codigo='" & txtCodigo.Text & "' " & _
            " AND mar_codigo='" & combo.SelectedValue & "' "
        SQL.Ejecutar(strSQL, "L", tipo)

    End If
End Sub

Private Sub combo_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles combo.SelectedIndexChanged
    Limpiar()
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
    End With
End Sub

```



```

        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
            .Cells(1, i).EntireRow.Font.Bold = True
            i += 1
        Next
        Dim k As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            i = 1
            For row = 0 To vsfg.Rows.Count - 1
                .Cells(i, k).Value = vsfg.Item(row, col)
                i += 1
            Next
            k += 1
        Next
    End With
End Sub
End Class

```

Listado de Vehiculos

```

Public Class frmListadoAutos
    Private strSQL As String
    Private SQL As New clsSQL
    Private Sub frmListadoAutos_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        cargarCombos()
        Limpiar()
    End Sub
    Public Sub cargarCombos()
        Dim ds0 As New DataTable, ds1 As New DataTable, ds2 As New
        DataTable, ds3 As New DataTable
        strSQL = " SELECT 1, '%' as codigo, 'Todos los Clientes' as nombre "
        & _
            " UNION " & _
            " SELECT 2, per_codigo as codigo, TRIM(CONCAT(per_apellido, '
', per_nombre)) as nombre " & _
            " FROM persona " & _
            " WHERE per_que='C' " & _
            " ORDER BY 1, nombre "

        SQL.Ejecutar(strSQL)
        ds2.Load(SQL.dataR, LoadOption.OverwriteChanges)
        comboCliente.DisplayMember = ds2.Columns.Item(2).Caption
        comboCliente.ValueMember = ds2.Columns.Item(1).Caption
        comboCliente.DataSource = ds2
        SQL.dataR.Close()

        strSQL = " SELECT 1, '%' as codigo, 'Todos los Modelos' as nombre " &
        _
            " UNION " & _

```

```

        " SELECT 2,mod_codigo as codigo,mod_nombre as nombre " & _
        " FROM modelo " & _
        " ORDER BY 1,nombre "

SQL.Ejecutar(strSQL)
ds0.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboModelo.DisplayMember = ds0.Columns.Item(2).Caption
comboModelo.ValueMember = ds0.Columns.Item(2).Caption
comboModelo.DataSource = ds0
SQL.dataR.Close()

strSQL = " SELECT 1,'% ' as codigo,'Todas las Marcas' as nombre " &
-
        " UNION " & _
        " SELECT 2,mar_codigo as codigo,mar_nombre as nombre " & _
        " FROM marca " & _
        " ORDER BY 1,nombre "

SQL.Ejecutar(strSQL)
ds1.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboMarca.DisplayMember = ds1.Columns.Item(2).Caption
comboMarca.ValueMember = ds1.Columns.Item(2).Caption
comboMarca.DataSource = ds1
SQL.dataR.Close()

strSQL = " SELECT 1,'% ' as codigo,'Todos los Colores' as nombre " &
-
        " UNION " & _
        " SELECT 2,col_codigo as codigo,col_nombre as nombre " & _
        " FROM color " & _
        " ORDER BY 1,nombre "

SQL.Ejecutar(strSQL)
ds3.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboColor.DisplayMember = ds3.Columns.Item(2).Caption
comboColor.ValueMember = ds3.Columns.Item(2).Caption
comboColor.DataSource = ds3
SQL.dataR.Close()

End Sub

Private Sub Limpiar()
    txtPlaca.Text = ""
    comboCliente.SelectedIndex = 0
    comboMarca.SelectedIndex = 0
    comboModelo.SelectedIndex = 0
    comboColor.SelectedIndex = 0
    txtAño.Text = ""
    Cargar()
    vsfg.Focus()
End Sub

Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub

Private Sub Cargar()

```

```

        Dim row As Integer, i As Integer

        strSQL = " SELECT auto.per_codigo,TRIM(CONCAT(per_apellido, '
',per_nombre)),
aut_codigo,mar_nombre,mod_nombre,col_nombre,aut_ano,aut_placa,aut_chasis,au
t_motor,aut_observacion " & _
            " FROM auto " & _
            " INNER JOIN modelo " & _
            " ON auto.mod_codigo=modelo.mod_codigo " & _
            " INNER JOIN marca " & _
            " ON marca.mar_codigo=modelo.mar_codigo " & _
            " LEFT JOIN color " & _
            " ON auto.col_codigo=color.col_codigo " & _
            " LEFT JOIN persona " & _
            " ON auto.per_codigo=persona.per_codigo " & _
            " AND persona.per_que='C' " & _
            " WHERE auto.per_codigo LIKE '" &
comboCliente.SelectedValue & "' " & _
            " AND marca.mar_nombre LIKE '" & txtMarca.Text &
"' " & _
            " AND color.col_nombre LIKE '" & txtColor.Text &
"' " & _
            " AND modelo.mod_nombre LIKE '" & txtModelo.Text &
"' " & _
            " AND aut_placa LIKE '%" & txtPlaca.Text & "%' " &
_
            " AND aut_ano LIKE '%" & txtAno.Text & "%' " & _
            " ORDER BY per_codigo,mar_nombre,mod_nombre "

        SQL.Ejecutar(strSQL)
        vsfg.Rows.Count = 1
        row = 1
        While SQL.dataR.Read()
            vsfg.Rows.Add()
            For i = 0 To SQL.dataR.FieldCount - 1
                vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
            Next
            row = row + 1
        End While
        SQL.dataR.Close()

        For i = 1 To vsfg.Rows.Count - 1
            vsfg.Item(i, 0) = CStr(i)
        Next
    End Sub

    Private Sub btnBuscar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnBuscar.Click

        Cargar()

    End Sub

    Private Sub comboMarca_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
comboMarca.SelectedIndexChanged

```

```

txtMarca.Text = comboMarca.SelectedValue

If txtMarca.Text = "Todas las Marcas" Then
    txtMarca.Text = "%"
Else
    txtMarca.Text = comboMarca.SelectedValue
End If
End Sub

Private Sub comboModelo_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
comboModelo.SelectedIndexChanged
    txtModelo.Text = comboModelo.SelectedValue

    If txtModelo.Text = "Todos los Modelos" Then
        txtModelo.Text = "%"
    Else
        txtModelo.Text = comboModelo.SelectedValue
    End If
End Sub

Private Sub comboColor_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
comboColor.SelectedIndexChanged
    txtColor.Text = comboColor.SelectedValue

    If txtColor.Text = "Todos los Colores" Then
        txtColor.Text = "%"
    Else
        txtColor.Text = comboColor.SelectedValue
    End If
End Sub

Private Sub txtAno_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtAno.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
            .Cells(1, i).EntireRow.Font.Bold = True
            i += 1
        Next
    End With
End Sub

```

```

        Dim k As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            i = 1
            For row = 0 To vsfg.Rows.Count - 1
                .Cells(i, k).Value = vsfg.Item(row, col)
                i += 1
            Next
            k += 1
        Next
    End With
End Sub
End Class

```

Cientes

```

Public Class frmClientes
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "el Cliente"
    PrivateCodigoPer As String = ""
    Private Sub frmClientes_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        cargarCombos()
        cargarCombo()
    End Sub
    Public Sub cargarCombos()
        Dim ds As New DataTable, ds1 As New DataTable
        Dim ds2 As New DataTable, ds3 As New DataTable

        Dim ds4 As New DataTable

        strSQL = " SELECT 'N' as codigo,'Natural' as nombre " & _
            " UNION " & _
            " SELECT 'J' as codigo,'Jurídica' as nombre "
        SQL.Ejecutar(strSQL)
        ds.Load(SQL.dataR, LoadOption.OverwriteChanges)
        comboTipo.DisplayMember = ds.Columns.Item(1).Caption
        comboTipo.ValueMember = ds.Columns.Item(0).Caption
        comboTipo.DataSource = ds
        SQL.dataR.Close()

        strSQL = " SELECT 1,'0' as codigo,'Seleccione uno...' as nombre,'"
ase_apellido,'" ase_nombre " & _
            " UNION " & _
            " SELECT 2,ase_codigo as codigo, CONCAT(ase_apellido,'"
',ase_nombre) as nombre,ase_apellido,ase_nombre " & _
            " FROM asesor " & _
            " ORDER BY ase_apellido,ase_nombre "
        SQL.Ejecutar(strSQL)
        ds3.Load(SQL.dataR, LoadOption.OverwriteChanges)
        comboAsesor.DisplayMember = ds3.Columns.Item(2).Caption
        comboAsesor.ValueMember = ds3.Columns.Item(2).Caption
        comboAsesor.DataSource = ds3
        SQL.dataR.Close()
    End Sub
End Class

```

```

End Sub

Private Sub Limpiar()
    txtCodigo.Text = ""
    txtCiudad.Text = ""
    comboAsesor.SelectedIndex = 0
    txtNombre.Text = ""
    txtApellido.Text = ""
    txtRuc.Text = ""
    txtDireccion.Text = ""
    txtTelefono.Text = ""
    txtCelular.Text = ""
    txtFax.Text = ""
    txtEMail.Text = ""
    txtActualiza.Text = ""
    txtIngresa.Text = ""
    txtWeb.Text = ""
    txtObservacion.Text = ""
    checkIva.Checked = True
    btnAuto.Enabled = False
    btnAceptar.Text = "&Aceptar"
    btnEliminar.Enabled = False
    txtRuc.Focus()
End Sub

Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub

Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
    Limpiar()
End Sub

Private Function controlDatos() As Boolean
    If Trim(txtRuc.Text) = "" Then

        MsgBox("Ingrese el ci/ruc para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtRuc.Focus()
        Return False

    ElseIf Trim(txtApellido.Text) = "" Then
        MsgBox("Ingrese el apellido/nombre de empresa para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtApellido.Focus()
        Return False
    Else
        Return True
    End If
End Function

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    txtIngresa.Text = "i"

```

```

txtActualiza.Text = "a"

If btnAceptar.Text = "&Aceptar" Then
    If controlDatos() = True Then
        If VerificarID(txtRuc.Text) Then
            manipularDatos("i")
            MsgBox("Se han ingresado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            cargarCombo()
            combo.SelectedValue = CodigoPer
        End If

        End If
        txtActualiza.Text = ""
        txtIngresa.Text = ""
        SQL.dataR.Close()
    Else
        If controlDatos() = True Then
            If VerificarID(txtRuc.Text) Then
                manipularDatos("a")
                MsgBox("Se han modificado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
                cargarCombo()
                combo.SelectedValue = CodigoPer
            End If
        End If
    End If
    SQL.dataR.Close()

End Sub

Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione " & Ventana & " que desea eliminar",
MsgBoxStyle.Information, Me.Text)
    Else
        Dim Cuantos As Integer = 0

        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
                " FROM auto " & _
                " WHERE per_codigo='" & txtCodigo.Text & "'"

        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
        SQL.dataR.Close()
        If Cuantos > 0 Then
            MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de Autos, no se puede
eliminar", MsgBoxStyle.Information, Me.Text)
            Exit Sub
        End If

        Cuantos = 0
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _

```

```

        " FROM proforma " & _
        " WHERE per_codigo='" & txtCodigo.Text & "'"

SQL.Ejecutar(strSQL)
SQL.dataR.Read()
Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
SQL.dataR.Close()
If Cuantos > 0 Then
    MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de Proformas, no se
puede eliminar", MsgBoxStyle.Information, Me.Text)
    Exit Sub
End If

Cuantos = 0
strSQL = " SELECT COALESCE(COUNT(*),0) " & _
        " FROM orden_trabajo " & _
        " WHERE per_codigo='" & txtCodigo.Text & "'"

SQL.Ejecutar(strSQL)
SQL.dataR.Read()
Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
SQL.dataR.Close()
If Cuantos > 0 Then
    MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de Orden de Trabajo, no
se puede eliminar", MsgBoxStyle.Information, Me.Text)
    Exit Sub
End If

Cuantos = 0
SQL.Ejecutar(strSQL)
SQL.dataR.Read()
Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
SQL.dataR.Close()
If Cuantos > 0 Then
    MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de Facturas, no se
puede eliminar", MsgBoxStyle.Information, Me.Text)
    Exit Sub
End If

If MsgBox("Está seguro que desea eliminar " & Ventana & " " &
txtNombre.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
    manipularDatos("e")
    MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)
    cargarCombo()
    combo.SelectedValue = CodigoPer
End If
End If
End Sub

```



```

Private Sub manipularDatos(ByVal tipo As String)
    If UCase(tipo) = "I" Then
        Dim codigo As String = "C00001"
        strSQL = " SELECT
cast(CONCAT('C',LPAD(COALESCE(MAX(RIGHT(per_codigo,LENGTH(per_codigo)-
1))+1,1),4,'0')) as char) as codigo " & _
        " FROM persona " & _
        " WHERE per_que='C' "

        SQL.Ejecutar(strSQL)
        If SQL.dataR.HasRows Then
            SQL.dataR.Read()
            codigo = SQL.dataR.GetValue(0).ToString
            SQL.dataR.Close()
        End If
        CodigoPer = codigo
        strSQL = "INSERT INTO
persona(per_que,per_codigo,per_tipo,per_ruc," & _
        "per_apellido,per_nombre,per_direccion,per_telefono,per_celular,per_fax," & _
        "per_ciudad,per_email,per_web,ase_nombre,per_observaciones) VALUES(" & _
        "'C','" & _
        codigo & "','" & _
        comboTipo.SelectedValue & "','" & _
        Trim(txtRuc.Text) & "','" & _
        Texto(txtApellido.Text) & "','" & _
        Texto(txtNombre.Text) & "','" & _
        Trim(txtDireccion.Text) & "','" & _
        Trim(txtTelefono.Text) & "','" & _
        Trim(txtCelular.Text) & "','" & _
        Trim(txtFax.Text) & "','" & _
        Trim(txtCiudad.Text) & "','" & _
        Trim(txtEMail.Text) & "','" & _
        Trim(txtWeb.Text) & "','" & _
        comboAsesor.SelectedValue & "','" & _
        Texto(txtObservacion.Text) & "')"
        SQL.Ejecutar(strSQL, "L", tipo)

    ElseIf UCase(tipo) = "E" Then
        strSQL = " DELETE FROM persona " & _
        " WHERE per_codigo='" & txtCodigo.Text & "' " & _
        " AND per_que='C' "
        SQL.Ejecutar(strSQL, "L", tipo)
        CodigoPer = txtCodigo.Text

    Else
        strSQL = " UPDATE persona SET " & _
        " per_tipo='" & comboTipo.SelectedValue & "','" & _
        " per_ruc='" & Trim(txtRuc.Text) & "','" & _
        " per_apellido='" & Texto(txtApellido.Text) & "','" & _
        " per_nombre='" & Texto(txtNombre.Text) & "','" & _
        " per_direccion='" & Trim(txtDireccion.Text) & "','" & _

```

```

        " per_telefono='" & Trim(txtTelefono.Text) & "'," & _
        " per_celular='" & Trim(txtCelular.Text) & "'," & _
        " per_fax='" & Trim(txtFax.Text) & "'," & _
        " per_ciudad='" & txtCiudad.Text & "'," & _
        " per_email='" & Trim(txtEMail.Text) & "'," & _
        " per_web='" & Trim(txtWeb.Text) & "'," & _
        " ase_nombre='" & comboAsesor.SelectedValue & "'," & _
        " per_observaciones='" & Texto(txtObservacion.Text) &
'"' & _
        " WHERE per_codigo='" & txtCodigo.Text & "' " & _
        " AND per_que='C' "
    SQL.Ejecutar(strSQL, "L", tipo)

    CodigoPer = txtCodigo.Text
End If
End Sub

Public Sub cargarCombo()
    strSQL = " SELECT 1,'% ' as codigo,'Nuevo Cliente...' as nombre " &
    " UNION " & _
    " SELECT 2,per_codigo as codigo,TRIM(CONCAT(per_apellido,'
',per_nombre)) as nombre " & _
    " FROM persona " & _
    " WHERE per_que='C' " & _
    " ORDER BY 1,nombre "
    SQL.Ejecutar(strSQL)
    Dim ds As New DataTable
    ds.Load(SQL.dataR, LoadOption.OverwriteChanges)
    combo.DisplayMember = ds.Columns.Item(2).Caption
    combo.ValueMember = ds.Columns.Item(1).Caption
    combo.DataSource = ds
    SQL.dataR.Close()
End Sub

Private Sub combo_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles combo.SelectedIndexChanged
    Limpiar()
    Cargar(combo.SelectedValue, "")
End Sub

Private Sub btnAuto_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAuto.Click
    If frmAutos.Visible Then
        frmAutos.Activate()
    Else
        frmAutos.MdiParent = mdiPrincipal
        frmAutos.Persona = txtCodigo.Text
        frmAutos.Show()
    End If
End Sub

Private Sub txtRuc_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtRuc.LostFocus

    Dim Cuantos As Integer = 0
    If combo.SelectedValue = "%" Then

```

```

If (txtActualiza.Text <> "a" Or txtIngresa.Text <> "i") Then
    strSQL = " SELECT COALESCE(COUNT(*),0) " & _
        " FROM persona " & _
        " WHERE per_ruc='" & Trim(txtRuc.Text) & "' " & _
        " AND per_que='C' "
    SQL.Ejecutar(strSQL)
    If SQL.dataR.HasRows Then
        Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
    End If
    SQL.dataR.Close()
    If Cuantos > 0 Then
        If MsgBox("Este CI/RUC ya está registrado, desea cargar
la información actual?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text)
= MsgBoxResult.Yes Then
            Cargar("%", Trim(txtRuc.Text))
        Else
            txtRuc.Text = ""
            txtRuc.Focus()
        End If
    End If
End If
End Sub
Private Sub Cargar(ByVal txt1 As String, ByVal txt2 As String)
    strSQL = " SELECT per_codigo,per_tipo,per_nombre,per_ruc, " & _
        "
per_direccion,per_telefono,per_fax,per_email,per_web,per_ciudad,per_observa
ciones,ase_nombre,per_apellido,per_celular " & _
        " FROM persona " & _
        " WHERE per_que='C' " & _
        " AND per_codigo='" & txt1 & "' " & _
        " OR per_ruc='" & txt2 & "' "
    SQL.Ejecutar(strSQL)

    If SQL.dataR.HasRows Then
        SQL.dataR.Read()
        comboTipo.SelectedIndex = IIf(SQL.dataR.GetValue(1) = "N", 0,
1)

        txtNombre.Text = SQL.dataR.GetValue(2)
        txtApellido.Text = Valor(SQL.dataR, 12)
        txtRuc.Text = SQL.dataR.GetValue(3)
        txtDireccion.Text = Valor(SQL.dataR, 4)
        txtTelefono.Text = Valor(SQL.dataR, 5)
        txtCelular.Text = Valor(SQL.dataR, 13)
        txtFax.Text = Valor(SQL.dataR, 6)
        txtEMail.Text = Valor(SQL.dataR, 7)
        txtWeb.Text = Valor(SQL.dataR, 8)
        txtCiudad.Text = Valor(SQL.dataR, 9)
        txtObservacion.Text = Valor(SQL.dataR, 10)
        comboAsesor.SelectedValue = SQL.dataR.GetValue(11)
        btnAceptar.Text = "&Modificar"
        btnEliminar.Enabled = True
        btnAuto.Enabled = True
        txtCodigo.Text = SQL.dataR.GetValue(0)
        SQL.dataR.Close()
        comboTipo.Focus()
    End If
End Sub

```

```

        End If
        SQL.dataR.Close()

    End Sub
    Private Sub txtRuc_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtRuc.KeyPress
        If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
            e.KeyChar = ""
        End If
    End Sub
    Private Sub txtTelefono_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtTelefono.KeyPress
        If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
            e.KeyChar = ""
        End If
    End Sub
    Private Sub txtCelular_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtCelular.KeyPress
        If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
            e.KeyChar = ""
        End If
    End Sub
    Private Sub txtFax_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtFax.KeyPress
        If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
            e.KeyChar = ""
        End If
    End Sub
End Class

```

Proveedores

```

Public Class frmProveedores
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "el Proveedor"
    Private Sub frmProveedores_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        cargarCombos()
        cargarCombo()
        Limpiar()
    End Sub
    Public Sub cargarCombos()
        Dim ds As New DataTable, ds1 As New DataTable
        Dim ds2 As New DataTable, ds3 As New DataTable
        strSQL = " SELECT 'N' as codigo,'Natural' as nombre " & _
            " UNION " & _
            " SELECT 'J' as codigo,'Jurídica' as nombre "
        SQL.Ejecutar(strSQL)
        ds.Load(SQL.dataR, LoadOption.OverwriteChanges)
    End Sub

```

```

        comboTipo.DisplayMember = ds.Columns.Item(1).Caption
        comboTipo.ValueMember = ds.Columns.Item(0).Caption
        comboTipo.DataSource = ds
        SQL.dataR.Close()
    End Sub
    Public Sub cargarCombo()
        strSQL = " SELECT 1,'% ' as codigo,'Nuevo Proveedor...' as nombre "
& _
        " UNION " & _
        " SELECT 2,per_codigo as codigo,TRIM(CONCAT(per_apellido,'
',per_nombre)) as nombre " & _
        " FROM persona " & _
        " WHERE per_que='P' " & _
        " ORDER BY 1,nombre "
        SQL.Ejecutar(strSQL)
        Dim ds As New DataTable
        ds.Load(SQL.dataR, LoadOption.OverwriteChanges)
        combo.DisplayMember = ds.Columns.Item(2).Caption
        combo.ValueMember = ds.Columns.Item(1).Caption
        combo.DataSource = ds
        SQL.dataR.Close()
    End Sub

    Private Sub Limpiar()
        comboTipo.SelectedIndex = 0
        txtNombre.Text = ""
        txtApellido.Text = ""
        txtRuc.Text = ""
        txtActualiza.Text = ""
        txtIngresa.Text = ""
        txtDireccion.Text = ""
        txtTelefono.Text = ""
        txtFax.Text = ""
        txtEMail.Text = ""
        txtActualiza.Text = ""
        txtCiudad.Text = ""
        txtWeb.Text = ""
        txtObservacion.Text = ""
        btnAceptar.Text = "&Aceptar"
        btnEliminar.Enabled = False
        txtRuc.Focus()
    End Sub

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
        Limpiar()
    End Sub

    Private Sub Cargar(ByVal txt1 As String, ByVal txt2 As String)
        strSQL = " SELECT per_codigo,per_tipo,per_nombre,per_ruc, " & _

```

```

"
per_direccion,per_telefono,per_fax,per_email,per_web,per_ciudad,per_observa
ciones,per_apellido " & _
" FROM persona " & _
" WHERE per_que='P' " & _
" AND per_codigo='" & txt1 & "' " & _
" OR per_ruc='" & txt2 & "' "
SQL.Ejecutar(strSQL)
If SQL.dataR.HasRows Then
    SQL.dataR.Read()

    comboTipo.SelectedIndex = IIf(SQL.dataR.GetValue(1) = "N", 0,
1)

    txtNombre.Text = SQL.dataR.GetValue(2)
    txtApellido.Text = Valor(SQL.dataR, 11)
    txtRuc.Text = SQL.dataR.GetValue(3)
    txtDireccion.Text = Valor(SQL.dataR, 4)
    txtTelefono.Text = Valor(SQL.dataR, 5)
    txtFax.Text = Valor(SQL.dataR, 6)
    txtEMail.Text = Valor(SQL.dataR, 7)
    txtWeb.Text = Valor(SQL.dataR, 8)
    txtCiudad.Text = Valor(SQL.dataR, 9)
    txtObservacion.Text = Valor(SQL.dataR, 10)
    btnAceptar.Text = "&Modificar"
    btnEliminar.Enabled = True
    txtCodigo.Text = SQL.dataR.GetValue(0)
    SQL.dataR.Close()
    comboTipo.Focus()
End If
SQL.dataR.Close()
End Sub

Private Function controlDatos() As Boolean
    If Trim(txtRuc.Text) = "" Then
        MsgBox("Ingrese el ci/ruc para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtRuc.Focus()
        Return False
    ElseIf Trim(txtApellido.Text) = "" Then
        MsgBox("Ingrese el apellido/nombre de empresa para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtApellido.Focus()
        Return False
    Else
        Return True
    End If
End Function

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    txtIngresa.Text = "i"
    txtActualiza.Text = "a"

    If btnAceptar.Text = "&Aceptar" Then
        If controlDatos() = True Then
            If VerificarID(txtRuc.Text) Then

```

```

        manipularDatos("i")
        MsgBox("Se han ingresado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
        cargarCombo()
        Limpiar()
    End If
End If
Else
    If controlDatos() = True Then
        If VerificarID(txtRuc.Text) Then
            manipularDatos("a")
            txtActualiza.Text = "a"
            MsgBox("Se han modificado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    End If
End If
SQL.dataR.Close()

End Sub

Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione " & Ventana & " que desea eliminar",
MsgBoxStyle.Information, Me.Text)
    Else
        If MsgBox("Está seguro que desea eliminar " & Ventana & " " &
txtNombre.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
            manipularDatos("e")
            MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)
            cargarCombo()
            Limpiar()
        End If
    End If
End Sub

Private Sub manipularDatos(ByVal tipo As String)
    If UCase(tipo) = "I" Then
        Dim codigo As String = "P00001"
        strSQL = " SELECT
cast(CONCAT('P',LPAD(COALESCE(MAX(RIGHT(per_codigo,LENGTH(per_codigo)-
1))+1,1),4,'0')) as char) as codigo " & _
" FROM persona " & _
" WHERE per_que='P' "
        SQL.Ejecutar(strSQL)
        If SQL.dataR.HasRows Then
            SQL.dataR.Read()
            codigo = SQL.dataR.GetValue(0).ToString
            SQL.dataR.Close()
        End If
    End If
End Sub

```

```

        strSQL = " INSERT INTO
persona(per_que,per_codigo,per_tipo,per_ruc," & _
"per_apellido,per_nombre,per_direccion,per_telefono,per_fax," & _
        "per_ciudad,per_email,per_web,per_observaciones)
VALUES(' " & _
        "P'," & _
        codigo & "','" & _
        comboTipo.SelectedValue & "','" & _
        Trim(txtRuc.Text) & "','" & _
        Texto(txtApellido.Text) & "','" & _
        Texto(txtNombre.Text) & "','" & _
        Trim(txtDireccion.Text) & "','" & _
        Trim(txtTelefono.Text) & "','" & _
        Trim(txtFax.Text) & "','" & _
        Texto(txtCiudad.Text) & "','" & _
        Trim(txtEMail.Text) & "','" & _
        Trim(txtWeb.Text) & "','" & _
        Texto(txtObservacion.Text) & "'" )"
SQL.Ejecutar(strSQL, "L", tipo)
SQL.dataR.Close()
ElseIf UCase(tipo) = "E" Then
    strSQL = " DELETE FROM persona " & _
        " WHERE per_codigo='" & txtCodigo.Text & "' " & _
        " AND per_que='P' "
    SQL.Ejecutar(strSQL, "L", tipo)
    SQL.dataR.Close()
Else
    strSQL = " UPDATE persona SET " & _
        " per_tipo='" & comboTipo.SelectedValue & "','" & _
        " per_ruc='" & Trim(txtRuc.Text) & "','" & _
        " per_apellido='" & Texto(txtApellido.Text) & "','" & _
        " per_nombre='" & Texto(txtNombre.Text) & "','" & _
        " per_direccion='" & Trim(txtDireccion.Text) & "','" & _
        " per_telefono='" & Trim(txtTelefono.Text) & "','" & _
        " per_fax='" & Trim(txtFax.Text) & "','" & _
        " per_ciudad='" & Texto(txtCiudad.Text) & "','" & _
        " per_email='" & Trim(txtEMail.Text) & "','" & _
        " per_web='" & Trim(txtWeb.Text) & "','" & _
        " per_observaciones='" & Texto(txtObservacion.Text) & _
        "'" & _
        " WHERE per_codigo='" & txtCodigo.Text & "' " & _
        " AND per_que='P' "
    SQL.Ejecutar(strSQL, "L", tipo)
    SQL.dataR.Close()
End If
End Sub

Private Sub txtRuc_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtRuc.LostFocus
    If (txtActualiza.Text <> "a" Or txtIngresa.Text <> "i") Then
        Dim Cuantos As Integer = 0
        If combo.SelectedValue = "%" Then
            strSQL = " SELECT COALESCE(COUNT(*),0) " & _
                " FROM persona " & _

```



```

        " WHERE per_ruc='" & Trim(txtRuc.Text) & "' " & _
        " AND per_que='P' "
SQL.Ejecutar(strSQL)
If SQL.dataR.HasRows Then
    Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
End If
SQL.dataR.Close()
If Cuantos > 0 Then
    If MsgBox("Este CI/RUC ya está registrado, desea cargar
la información actual?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text)
= MsgBoxResult.Yes Then
        Cargar("%", Trim(txtRuc.Text))
    Else
        txtRuc.Text = ""
        txtRuc.Focus()
    End If
End If
End If
End If

End Sub

Private Sub combo_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles combo.SelectedIndexChanged
    Limpiar()
    Cargar(combo.SelectedValue, "")
End Sub

Private Sub txtTelefono_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtTelefono.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub txtFax_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtFax.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub txtRuc_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtRuc.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

End Sub

End Class

```

Listado Clientes

```

Public Class frmListadoClientes
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "el Cliente"

```

```

Private htCategoria As New Hashtable
Private htAsesor As New Hashtable
Private htCiudad As New Hashtable
Private Sub frmListadoClientes_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load
    SQL.Inicializar(ConnMaster)
    cargarCombos()
    Limpiar()
End Sub
Public Sub cargarCombos()
    Dim ds0 As New DataTable, ds1 As New DataTable
    strSQL = " SELECT 1, '%' as codigo, 'Todos los Asesores' as nombre "
& _
        " UNION " & _
        " SELECT 2, cast(ase_codigo as char) as
codigo, CONCAT(ase_apellido, ' ', ase_nombre) as nombre " & _
        " FROM asesor " & _
        " ORDER BY 1, nombre "
    SQL.Ejecutar(strSQL)
    ds1.Load(SQL.dataR, LoadOption.OverwriteChanges)
    comboAsesor.DisplayMember = ds1.Columns.Item(2).Caption
    comboAsesor.ValueMember = ds1.Columns.Item(2).Caption
    comboAsesor.DataSource = ds1
    SQL.dataR.Close()

    htCiudad.Clear()
    htCategoria.Clear()
    htAsesor.Clear()
    strSQL = " SELECT ase_codigo as codigo, CONCAT(ase_apellido, '
', ase_nombre) as nombre " & _
        " FROM asesor " & _
        " ORDER BY nombre "
    SQL.Ejecutar(strSQL)
    While SQL.dataR.Read
        htAsesor.Add(Valor(SQL.dataR, 0), Valor(SQL.dataR, 1))
    End While
    SQL.dataR.Close()
    strSQL = " SELECT ciu_codigo as codigo, CONCAT(ciu_nombre, '
', pai_nombre) as nombre " & _
        " FROM ciudad " & _
        " INNER JOIN pais " & _
        " ON pais.pai_codigo=ciudad.pai_codigo " & _
        " ORDER BY nombre "
    SQL.Ejecutar(strSQL)
    While SQL.dataR.Read
        htCiudad.Add(Valor(SQL.dataR, 0), Valor(SQL.dataR, 1))
    End While
    SQL.dataR.Close()
End Sub

Private Sub Limpiar()
    txtRucFiltro.Text = ""
    txtNombreFiltro.Text = ""
    comboAsesor.SelectedIndex = 0
    Cargar()

```

```

        vsfg.Focus()
    End Sub

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
        Limpiar()
    End Sub

    Private Sub Cargar()
        Dim row As Integer, i As Integer

        strSQL = " SELECT
per_codigo,IF(per_tipo='N','Natural','Jurídico'),per_apellido,per_nombre,pe
r_ruc," & _
        "
per_direccion,per_telefono,per_celular,per_fax,per_ciudad,per_email,per_web
,ase_nombre,per_observaciones " & _
        " FROM persona " & _
        " WHERE per_que='C' " & _
        " AND (per_nombre LIKE '%" & txtNombreFiltro.Text & "%' OR
per_apellido LIKE '%" & txtNombreFiltro.Text & "%') " & _
        " AND per_ruc LIKE '%" & txtRucFiltro.Text & "%' " & _
        " AND ase_nombre LIKE '%" & txtAsesor.Text & "%' " & _
        " ORDER BY per_apellido,per_nombre "
        SQL.Ejecutar(strSQL)
        vsfg.Rows.Count = 1
        row = 1
        While SQL.dataR.Read()
            vsfg.Rows.Add()
            For i = 0 To SQL.dataR.FieldCount - 1
                vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
            Next
            row = row + 1
        End While
        SQL.dataR.Close()

        For i = 1 To vsfg.Rows.Count - 1
            vsfg.Item(i, 0) = CStr(i)
        Next

    End Sub

    Private Sub btnBuscar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnBuscar.Click
        If txtRucFiltro.Text <> "" Then
            If VerificarID(txtRucFiltro.Text) Then
                Cargar()
            End If
        Else
            Cargar()
        End If
    End Sub

```

```

End Sub

Private Sub comboAsesor_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
comboAsesor.SelectedIndexChanged

    txtAsesor.Text = comboAsesor.SelectedValue

    If txtAsesor.Text = "Todos los Asesores" Then
        txtAsesor.Text = "%"
    Else
        txtAsesor.Text = comboAsesor.SelectedValue
    End If

End Sub

Private Sub txtRucFiltro_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtRucFiltro.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
            .Cells(1, i).EntireRow.Font.Bold = True
            i += 1
        Next
        Dim k As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            i = 1
            For row = 0 To vsfg.Rows.Count - 1
                .Cells(i, k).Value = vsfg.Item(row, col)
                i += 1
            Next
            k += 1
        Next
    End With
End Sub
End Class

```

Listado Proveedores

```

Public Class frmListadoProveedores
    Private strSQL As String

```

```

Private SQL As New clsSQL
Private Const Ventana As String = "el Proveedor"
Private htCategoria As New Hashtable
Private htCiudad As New Hashtable
Private Sub frmListadoProveedores_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load
    SQL.Inicializar(ConnMaster)
    cargarCombos()
    Limpiar()
End Sub
Public Sub cargarCombos()
    Dim ds0 As New DataTable, ds1 As New DataTable

    htCiudad.Clear()
    htCategoria.Clear()
    strSQL = " SELECT ciu_codigo as codigo,CONCAT(ciu_nombre,'
',pai_nombre) as nombre " & _
            " FROM ciudad " & _
            " INNER JOIN pais " & _
            " ON pais.pai_codigo=ciudad.pai_codigo " & _
            " ORDER BY nombre "
    SQL.Ejecutar(strSQL)
    While SQL.dataR.Read
        htCiudad.Add(Valor(SQL.dataR, 0), Valor(SQL.dataR, 1))
    End While
    SQL.dataR.Close()
End Sub

Private Sub Limpiar()
    txtRucFiltro.Text = ""
    txtNombreFiltro.Text = ""
    Cargar()
    vsfg.Focus()
End Sub

Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub

Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs)
    Limpiar()
End Sub

Private Sub Cargar()
    Dim row As Integer, i As Integer
    strSQL = " SELECT
per_codigo,IF(per_tipo='N','Natural','Jurídico'),per_apellido,per_nombre,pe
r_ruc, " & _
            "
per_direccion,per_telefono,per_fax,per_ciudad,per_email,per_web,per_observa
ciones " & _
            " FROM persona " & _
            " WHERE per_que='P' " & _

```

```

        " AND (per_nombre LIKE '%" & txtNombreFiltro.Text & '%" OR
per_apellido LIKE '%" & txtNombreFiltro.Text & '%" ) " & _
        " AND per_ruc LIKE '%" & txtRucFiltro.Text & '%" " & _
        " ORDER BY per_apellido,per_nombre "
SQL.Ejecutar(strSQL)
vsfg.Rows.Count = 1
row = 1
While SQL.dataR.Read()
    vsfg.Rows.Add()
    For i = 0 To SQL.dataR.FieldCount - 1
        vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
    Next
    row = row + 1
End While
SQL.dataR.Close()

For i = 1 To vsfg.Rows.Count - 1
    vsfg.Item(i, 0) = CStr(i)
Next
End Sub

Private Sub btnBuscar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnBuscar.Click
    If txtRucFiltro.Text <> "" Then
        If VerificarID(txtRucFiltro.Text) Then
            Cargar()
        End If
    Else
        Cargar()
    End If
End Sub

Private Sub txtRucFiltro_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtRucFiltro.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
            .Cells(1, i).EntireRow.Font.Bold = True
            i += 1
        Next
        Dim k As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1

```

```

        i = 1
        For row = 0 To vsfg.Rows.Count - 1
            .Cells(i, k).Value = vsfg.Item(row, col)
            i += 1
        Next
        k += 1
    Next
End With
End Sub
End Class

```

Repuestos

```

Public Class frmRepuestos
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "el Repuesto"
    Private Sub frmRepuestos_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        cargarCombos()
        Limpiar()
    End Sub
    Public Sub cargarCombos()
        Dim ds1 As New DataTable
        Dim ds2 As New DataTable
        strSQL = " SELECT 1,'0' as codigo,'Seleccione uno...' as nombre " &
        " UNION " & _
        " SELECT 2,gru_pro_codigo as codigo,gru_pro_nombre as nombre " & _
        " FROM grupo_producto " & _
        " ORDER BY 1,codigo "
        SQL.Ejecutar(strSQL)
        ds2.Load(SQL.dataR, LoadOption.OverwriteChanges)
        comboGrupo.DisplayMember = ds2.Columns.Item(2).Caption
        comboGrupo.ValueMember = ds2.Columns.Item(1).Caption
        comboGrupo.DataSource = ds2
        SQL.dataR.Close()
    End Sub

    Private Sub Limpiar()
        comboGrupo.SelectedIndex = 0
        txtCodigo.Text = ""
        txtMarca.Text = ""
        txtNombre.Text = ""
        txtCosto.Text = Numero("0,00")
        txtDescripcion.Text = ""
        txtMax.Text = ""
        txtMin.Text = ""
        txtUbicacion.Text = ""
        checkPuntoPedido.Checked = False
        btnAceptar.Text = "&Aceptar"
        btnEliminar.Enabled = False
        txtCodigo.Enabled = True
        Cargar()
    End Sub

```

```

        txtCodigo.Focus()
    End Sub

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
        Limpiar()
    End Sub

    Private Sub Cargar()
        Dim row As Integer, i As Integer
        strSQL = " SELECT
pro_codigo,pro_nombre,pro_marca,producto.gru_pro_codigo,gru_pro_nombre,pro_
costo,pro_ubicacion,pro_max,pro_min,pro_pto_pedido,pro_descripcion " & _
            " FROM producto " & _
            " LEFT JOIN grupo_producto " & _
            " ON grupo_producto.gru_pro_codigo=producto.gru_pro_codigo
" & _
            " WHERE pro_tipo='R' "
        SQL.Ejecutar(strSQL)
        vsfg.Rows.Count = 1
        row = 1
        While SQL.dataR.Read()
            vsfg.Rows.Add()
            For i = 0 To SQL.dataR.FieldCount - 1
                vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
            Next
            row = row + 1
        End While
        SQL.dataR.Close()

        For i = 1 To vsfg.Rows.Count - 1
            vsfg.Item(i, 0) = CStr(i)
        Next
    End Sub

    Private Function controlDatos() As Boolean
        If Trim(txtCodigo.Text) = "" Then
            MsgBox("Ingrese el código para " & Ventana,
MsgBoxStyle.Information, Me.Text)
            txtCodigo.Focus()
            Return False
        ElseIf Trim(txtNombre.Text) = "" Then
            MsgBox("Ingrese el nombre para " & Ventana,
MsgBoxStyle.Information, Me.Text)
            txtNombre.Focus()
            Return False
        ElseIf btnAceptar.Text = "&Aceptar" Then
            Dim Cuantos As Integer = 0
            strSQL = " SELECT COALESCE(COUNT(pro_codigo),0) " & _
                " FROM producto " & _
                " WHERE pro_codigo='" & txtCodigo.Text & "' "

```



```

SQL.Ejecutar(strSQL)
If SQL.dataR.HasRows Then
    SQL.dataR.Read()
    Cuantos = SQL.dataR.GetValue(0).ToString
    SQL.dataR.Close()
End If

If Cuantos > 0 Then
    MsgBox("El código ingresado ya existe, verifique la
información", MsgBoxStyle.Information, Me.Text)
    Return False
Else
    Return True
End If
Else
    Return True
End If
End Function

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If btnAceptar.Text = "&Aceptar" Then
        If controlDatos() = True Then
            manipularDatos("i")
            MsgBox("Se han ingresado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    Else
        If controlDatos() = True Then
            manipularDatos("a")
            MsgBox("Se han modificado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    End If
End Sub

Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione " & Ventana & " que desea eliminar",
MsgBoxStyle.Information, Me.Text)
    Else
        Dim Cuantos As Integer = 0
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
                " FROM detalle_proforma " & _
                " WHERE pro_codigo='" & txtCodigo.Text & "'"
        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
        SQL.dataR.Close()
        If Cuantos > 0 Then
            MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de Proformas, no se
puede eliminar", MsgBoxStyle.Information, Me.Text)
        End If
    End If
End Sub

```

```

        Exit Sub
    End If
    Cuantos = 0
    strSQL = " SELECT COALESCE(COUNT(*),0) " & _
            " FROM detalle_orden_trabajo " & _
            " WHERE pro_codigo='" & txtCodigo.Text & "'"
    SQL.Ejecutar(strSQL)
    SQL.dataR.Read()
    Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
    SQL.dataR.Close()
    If Cuantos > 0 Then
        MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de Orden de Trabajo, no
se puede eliminar", MsgBoxStyle.Information, Me.Text)
        Exit Sub
    End If
    Cuantos = 0
    strSQL = " SELECT COALESCE(COUNT(*),0) " & _
            " FROM detalle_facturacion " & _
            " WHERE pro_codigo='" & txtCodigo.Text & "'"
    SQL.Ejecutar(strSQL)
    SQL.dataR.Read()
    Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
    SQL.dataR.Close()
    If Cuantos > 0 Then
        MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de Facturas, no se
puede eliminar", MsgBoxStyle.Information, Me.Text)
        Exit Sub
    End If

    If MsgBox("Está seguro que desea eliminar " & Ventana & " " &
txtNombre.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
        manipularDatos("e")
        MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)
        Limpiar()
    End If
End If
End Sub

Private Sub manipularDatos(ByVal tipo As String)
    Dim costo As Double

    If UCase(tipo) = "I" Then
        strSQL = " INSERT INTO
producto(pro_tipo,uni_codigo,pro_codigo," & _
"pro_nombre,pro_marca,gru_pro_codigo,pro_costo,pro_ubicacion,pro_max,pro_mi
n,pro_pto_pedido,pro_num_mecanico,pro_tiempo,pro_descripcion) VALUES(' " & _
"R','U',' " & _
Texto(txtCodigo.Text) & "',' " & _
Texto(txtNombre.Text) & "',' " & _
Texto(txtMarca.Text) & "',' " & _
comboGrupo.SelectedValue & "',' " & _

```

```

        NBDD(txtCosto.Text) & "','" & _
        Texto(txtUbicacion.Text) & "','" & _
        Texto(txtMax.Text) & "','" & _
        Texto(txtMin.Text) & "','" & _
        Math.Abs(CInt(checkPuntoPedido.Checked)) &
"'','0','0',''" & _
        Texto(txtDescripcion.Text) & "'') "
    SQL.Ejecutar(strSQL, "L", tipo)

    strSQL = " INSERT INTO existencia(pro_codigo,exi_cantidad)
VALUES('" & _
        Texto(txtCodigo.Text) & "','0) "
    SQL.Ejecutar(strSQL, "L", tipo)

ElseIf UCase(tipo) = "E" Then
    strSQL = " DELETE FROM producto " & _
        " WHERE pro_codigo='" & txtCodigo.Text & "' " & _
        " AND pro_tipo='R' "
    SQL.Ejecutar(strSQL, "L", tipo)
    strSQL = " DELETE FROM existencia " & _
        " WHERE pro_codigo='" & txtCodigo.Text & "' "
    SQL.Ejecutar(strSQL, "L", tipo)

Else

    costo = CDb1(txtCosto.Text)

    strSQL = " UPDATE producto SET " & _
        " uni_codigo='U'," & _
        " pro_nombre='" & Texto(txtNombre.Text) & "','" & _
        " pro_marca='" & Texto(txtMarca.Text) & "','" & _
        " gru_pro_codigo='" & comboGrupo.SelectedValue & "','" &
-
        " pro_costo='" & NBDD(costo) & "','" & _
        " pro_ubicacion='" & Texto(txtUbicacion.Text) & "','" &
-
        " pro_max='" & Texto(txtMax.Text) & "','" & _
        " pro_min='" & Texto(txtMin.Text) & "','" & _
        " pro_pto_pedido='" &
Math.Abs(CInt(checkPuntoPedido.Checked)) & "','" & _
        " pro_descripcion='" & Texto(txtDescripcion.Text) & "' "
& _
        " WHERE pro_codigo='" & txtCodigo.Text & "' " & _
        " AND pro_tipo='R' "
    SQL.Ejecutar(strSQL, "L", tipo)
End If
End Sub

Private Sub txtCosto_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtCosto.LostFocus
    Dim costo As Double
    costo = CDb1(txtCosto.Text)

    txtCosto.Text = FormatNumber(Numero(costo), 2)
End Sub

```

```

Private Sub datosRepuestos()
    Dim row As Integer
    row = vsfg.Row

    txtCodigo.Text = vsfg.Item(row, 1)
    txtNombre.Text = vsfg.Item(row, 2)
    txtMarca.Text = vsfg.Item(row, 3)
    comboGrupo.SelectedValue = vsfg.Item(row, 4)
    txtCosto.Text = vsfg.Item(row, 6)
    txtUbicacion.Text = vsfg.Item(row, 7)
    txtMax.Text = FormatNumber(Numero(vsfg.Item(row, 8)), 0)
    txtMin.Text = FormatNumber(Numero(vsfg.Item(row, 9)), 0)
    txtDescripcion.Text = vsfg.Item(row, 11)

    btnAceptar.Text = "&Modificar"
    txtCodigo.Enabled = False
    btnEliminar.Enabled = True

    strSQL = " SELECT exi_cantidad " & _
            " FROM existencia " & _
            "WHERE pro_codigo='" & txtCodigo.Text & "' "
    SQL.Ejecutar(strSQL)
    If SQL.dataR.HasRows Then
        SQL.dataR.Read()
        txtPtoServ.Text = SQL.dataR.GetValue(0).ToString
        SQL.dataR.Close()
    End If
    If CInt(txtPtoServ.Text) <= (CInt(txtMin.Text) + 1) Then
        checkPuntoPedido.Checked = True
        manipularDatos("a")

    Else
        checkPuntoPedido.Checked = False
        manipularDatos("a")
    End If
End Sub

Private Sub vsfg_DoubleClick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
    datosRepuestos()
End Sub

Public Function txtNumerico(ByVal txtControl As TextBox, ByVal caracter
As Char, ByVal decimales As Boolean) As Boolean
    If (Char.IsNumber(caracter, 0) = True) Or caracter =
Convert.ToChar(8) Or caracter = "," Then
        If caracter = "," Then
            If decimales = True Then
                If txtControl.Text.IndexOf(",") <> -1 Then Return True
            Else : Return True
            End If
        End If
        Return False
    Else
        Return True
    End If
End Function

```

```

    End If
End Function
'Validacion para numero decimales en mano de obra
Private Sub txtCosto_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtCosto.KeyPress
    e.Handled = txtNumerico(txtCosto, e.KeyChar, True)
End Sub
Private Sub txtMax_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtMax.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub
Private Sub txtMin_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtMin.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub
Private Sub txtMin_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtMin.LostFocus
    If txtMin.Text = "" Then
    ElseIf CInt(txtMin.Text) > (CInt(txtMax.Text)) Then
        MsgBox("El valor " & txtMin.Text & " no puede sobrepasar el
máximo permitido", MsgBoxStyle.Information, Me.Text)
        txtMin.Text = ""
        txtMin.Focus()
    Else
    End If
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
            .Cells(1, i).EntireRow.Font.Bold = True
            i += 1
        Next

        Dim k As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            i = 1
            For row = 0 To vsfg.Rows.Count - 1
                .Cells(i, k).Value = vsfg.Item(row, col)
                i += 1
            Next
            k += 1
        Next
    End With
End Sub

```

```

        Next
    End With
End Sub
End Class

```

Servicios

```

Public Class frmServicios
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "el Servicio"
    Private Sub frmServicios_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        cargarCombos()
        Limpiar()
    End Sub
    Public Sub cargarCombos()
        Dim ds1 As New DataTable
        Dim ds2 As New DataTable

        strSQL = " SELECT 1,'0' as codigo,'Seleccione uno...' as nombre " &
        " UNION " & _
        " SELECT 2,gru_pro_codigo as codigo,gru_pro_nombre as
nombre " & _
        " FROM grupo_producto " & _
        " ORDER BY 1,codigo "
        SQL.Ejecutar(strSQL)
        ds2.Load(SQL.dataR, LoadOption.OverwriteChanges)
        comboGrupo.DisplayMember = ds2.Columns.Item(2).Caption
        comboGrupo.ValueMember = ds2.Columns.Item(1).Caption
        comboGrupo.DataSource = ds2
        SQL.dataR.Close()
    End Sub

    Private Sub Limpiar()
        comboGrupo.SelectedIndex = 0
        txtCodigo.Text = ""
        txtNombre.Text = ""
        txtMarca.Text = ""
        txtCosto.Text = Numero("0")
        txtDescripcion.Text = ""
        txtMecanico.Text = ""
        txtTiempoEstimado.Text = ""
        btnAceptar.Text = "&Aceptar"
        btnEliminar.Enabled = False
        txtCodigo.Enabled = True
        Cargar()
        txtCodigo.Focus()
    End Sub

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

```

```

End Sub

Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
    Limpiar()
End Sub

Private Sub Cargar()
    Dim row As Integer, i As Integer
    strSQL = " SELECT
pro_codigo,pro_nombre,pro_marca,producto.gru_pro_codigo,gru_pro_nombre,pro_
costo,pro_num_mecanico,pro_tiempo,pro_descripcion " & _
        " FROM producto " & _
        " LEFT JOIN grupo_producto " & _
        " ON grupo_producto.gru_pro_codigo=producto.gru_pro_codigo
" & _
        " WHERE pro_tipo='S' "
    SQL.Ejecutar(strSQL)
    vsfg.Rows.Count = 1
    row = 1
    While SQL.dataR.Read()
        vsfg.Rows.Add()
        For i = 0 To SQL.dataR.FieldCount - 1
            vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
        Next
        row = row + 1
    End While
    SQL.dataR.Close()

    For i = 1 To vsfg.Rows.Count - 1
        vsfg.Item(i, 0) = CStr(i)
    Next
End Sub

Private Function controlDatos() As Boolean
    If Trim(txtCodigo.Text) = "" Then
        MsgBox("Ingrese el código para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtCodigo.Focus()
        Return False
    ElseIf Trim(txtNombre.Text) = "" Then
        MsgBox("Ingrese el nombre para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtNombre.Focus()
        Return False
    ElseIf btnAceptar.Text = "&Aceptar" Then
        Dim Cuantos As Integer = 0
        strSQL = " SELECT COALESCE(COUNT(pro_codigo),0) " & _
            " FROM producto " & _
            " WHERE pro_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL)
        If SQL.dataR.HasRows Then
            SQL.dataR.Read()
            Cuantos = SQL.dataR.GetValue(0).ToString
            SQL.dataR.Close()
        End If
    End If
End Function

```

```

        If Cuantos > 0 Then
            MsgBox("El código ingresado ya existe, verifique la
información", MsgBoxStyle.Information, Me.Text)
            Return False
        Else
            Return True
        End If
    Else
        Return True
    End If
End Function

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If btnAceptar.Text = "&Aceptar" Then
        If controlDatos() = True Then
            manipularDatos("i")
            MsgBox("Se han ingresado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    Else
        If controlDatos() = True Then
            manipularDatos("a")
            MsgBox("Se han modificado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    End If
End Sub

Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione " & Ventana & " que desea eliminar",
MsgBoxStyle.Information, Me.Text)
    Else
        Dim Cuantos As Integer = 0
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
            " FROM detalle_proforma " & _
            " WHERE pro_codigo='" & txtCodigo.Text & "'"
        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
        SQL.dataR.Close()
        If Cuantos > 0 Then
            MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de Proformas, no se
puede eliminar", MsgBoxStyle.Information, Me.Text)
            Exit Sub
        End If
        Cuantos = 0
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
            " FROM detalle_orden_trabajo " & _
            " WHERE pro_codigo='" & txtCodigo.Text & "'"
    End If
End Sub

```



```

SQL.Ejecutar(strSQL)
SQL.dataR.Read()
Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
SQL.dataR.Close()
If Cuantos > 0 Then
    MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de Orden de Trabajo, no
se puede eliminar", MsgBoxStyle.Information, Me.Text)
    Exit Sub
End If
Cuantos = 0
strSQL = " SELECT COALESCE(COUNT(*),0) " & _
        " FROM detalle_facturacion " & _
        " WHERE pro_codigo='" & txtCodigo.Text & "'"
SQL.Ejecutar(strSQL)
SQL.dataR.Read()
Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
SQL.dataR.Close()
If Cuantos > 0 Then
    MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de Facturas, no se
puede eliminar", MsgBoxStyle.Information, Me.Text)
    Exit Sub
End If

If MsgBox("Está seguro que desea eliminar " & Ventana & " " &
txtNombre.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
    manipularDatos("e")
    MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)
    Limpiar()
End If
End If
End Sub

Private Sub manipularDatos(ByVal tipo As String)
    Dim costo As Double

    If UCase(tipo) = "I" Then
        strSQL = " INSERT INTO
producto(pro_tipo,uni_codigo,pro_codigo," & _
        "pro_nombre,pro_marca,gru_pro_codigo,pro_costo,pro_ubicacion,pro_max,pro_mi
n,pro_pto_pedido,pro_num_mecanico,pro_tiempo,pro_descripcion) VALUES(' " & _
        "S','U',' " & _
        Texto(txtCodigo.Text) & "',' " & _
        Texto(txtNombre.Text) & "',' " & _
        Texto(txtMarca.Text) & "',' " & _
        comboGrupo.SelectedValue & "',' " & _
        NBDD(txtCosto.Text) & "','','0','0','0',' " & _
        Texto(txtMecanico.Text) & "',' " & _
        Texto(txtTiempoEstimado.Text) & "',' " & _
        Texto(txtDescripcion.Text) & "'" )"
SQL.Ejecutar(strSQL, "I", tipo)

```

```

ElseIf UCase(tipo) = "E" Then
    strSQL = " DELETE FROM producto " & _
            " WHERE pro_codigo='" & txtCodigo.Text & "' " & _
            " AND pro_tipo='S' "
    SQL.Ejecutar(strSQL, "L", tipo)

Else
    costo = CDb1(txtCosto.Text)

    strSQL = " UPDATE producto SET " & _
            " uni_codigo='U'," & _
            " pro_nombre='" & Texto(txtNombre.Text) & "'," & _
            " pro_marca='" & Texto(txtMarca.Text) & "'," & _
            " gru_pro_codigo='" & comboGrupo.SelectedValue & "'," &
-
            " pro_costo='" & NBDD(costo) & "'," & _
& _
            " pro_num_mecanico='" & Texto(txtMecanico.Text) & "'," &
-
            " pro_tiempo='" & Texto(txtTiempEstimado.Text) & "'," &
& _
            " pro_descripcion='" & Texto(txtDescripcion.Text) & "' "
            " WHERE pro_codigo='" & txtCodigo.Text & "' " & _
            " AND pro_tipo='S' "
    SQL.Ejecutar(strSQL, "L", tipo)
End If
End Sub

Private Sub txtCosto_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtCosto.LostFocus
    Dim costo As Double
    costo = CDb1(txtCosto.Text)
    txtCosto.Text = FormatNumber(Numero(costo), 2)
End Sub

Private Sub datosRepuestos()
    Dim row As Integer
    row = vsfg.Row

    txtCodigo.Text = vsfg.Item(row, 1)
    txtNombre.Text = vsfg.Item(row, 2)
    txtMarca.Text = vsfg.Item(row, 3)
    comboGrupo.SelectedValue = vsfg.Item(row, 4)
    txtCosto.Text = FormatNumber(Numero(vsfg.Item(row, 6)), 2)
    txtMecanico.Text = FormatNumber(Numero(vsfg.Item(row, 7)), 0)
    txtTiempEstimado.Text = FormatNumber(Numero(vsfg.Item(row, 8)), 0)
    txtDescripcion.Text = vsfg.Item(row, 9)
    btnAceptar.Text = "&Modificar"
    txtCodigo.Enabled = False
    btnEliminar.Enabled = True
End Sub

Private Sub vsfg_DoubleClick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
    datosRepuestos()
End Sub

```

```

    Public Function txtNumerico(ByVal txtControl As TextBox, ByVal caracter
As Char, ByVal decimales As Boolean) As Boolean
        If (Char.IsNumber(caracter, 0) = True) Or caracter =
Convert.ToChar(8) Or caracter = "," Then
            If caracter = "," Then
                If decimales = True Then
                    If txtControl.Text.IndexOf(",") <> -1 Then Return True
                Else : Return True
            End If
        End If
        Return False
    Else
        Return True
    End If
End Function
'Validacion para numero decimales en mano de obra
Private Sub txtCosto_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtCosto.KeyPress
    e.Handled = txtNumerico(txtCosto, e.KeyChar, True)
End Sub
Private Sub txtMecanico_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtMecanico.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub
Private Sub txtTiempEstimado_KeyPress(ByVal sender As Object, ByVal e
As System.Windows.Forms.KeyPressEventArgs) Handles
txtTiempEstimado.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub
Private Sub txtTiempEstimado_LostFocus(ByVal sender As Object, ByVal e
As System.EventArgs) Handles txtTiempEstimado.LostFocus
    If CInt(txtTiempEstimado.Text) > 60 Then
        MsgBox("El valor " & txtTiempEstimado.Text & " no está
permitido", MsgBoxStyle.Information, Me.Text)
        txtTiempEstimado.Text = ""
        txtTiempEstimado.Focus()
    End If
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
        Next
    End With
End Sub

```

```

        .Cells(1, i).EntireRow.Font.Bold = True
        i += 1
    Next
    Dim k As Integer = 1
    For col = 0 To vsfg.Cols.Count - 1
        i = 1
        For row = 0 To vsfg.Rows.Count - 1
            .Cells(i, k).Value = vsfg.Item(row, col)
            i += 1
        Next
        k += 1
    Next
End With
End Sub
End Class

```

Grupos Repuestos/Servicios

```

Public Class frmGruposProductos
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "el Grupo"
    Private Sub frmGruposProductos_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        Limpiar()
    End Sub
    Private Sub Limpiar()
        txtCodigo.Text = ""
        txtCodigo.Enabled = True
        comboCodigo.Enabled = False
        radioGrupo.Checked = True
        txtNombre.Text = ""
        txtDescripcion.Text = ""
        btnAceptar.Text = "&Aceptar"
        cargarCombo()
        cargarCodigo()
        txtCodigo.Focus()
        btnEliminar.Enabled = False
        Cargar()
    End Sub

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
        Limpiar()
    End Sub

    Private Sub Cargar()
        Dim row As Integer, i As Integer

```

```

strSQL = " SELECT gru_pro_codigo,gru_pro_nombre,gru_pro_descripcion
" & _
        " FROM grupo_producto " & _
        " ORDER BY gru_pro_codigo "
SQL.Ejecutar(strSQL)
vsfg.Rows.Count = 1
row = 1
While SQL.dataR.Read()
    vsfg.Rows.Add()
    For i = 0 To SQL.dataR.FieldCount - 1
        vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
    Next
    row = row + 1
End While
SQL.dataR.Close()

For i = 1 To vsfg.Rows.Count - 1
    vsfg.Item(i, 0) = CStr(i)
Next
End Sub

Private Sub datosColor()
    Dim row As Integer
    row = vsfg.Row

    txtCodigo.Text = vsfg.Item(row, 1)
    txtNombre.Text = vsfg.Item(row, 2)
    txtDescripcion.Text = vsfg.Item(row, 3)

    btnAceptar.Text = "&Modificar"
    txtCodigo.Enabled = False
    btnEliminar.Enabled = True
End Sub

Private Sub vsfg_DoubleClick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
    datosColor()
End Sub

Private Function controlDatos() As Boolean
    If Trim(txtNombre.Text) = "" Then
        MsgBox("Ingrese un nombre para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtNombre.Focus()
        Return False
    Else
        Return True
    End If
End Function

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If btnAceptar.Text = "&Aceptar" Then
        If controlDatos() = True Then
            manipularDatos("i")
        End If
    End If
End Sub

```

```

        MsgBox("Se han ingresado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
        Limpiar()
    End If
Else
    If controlDatos() = True Then
        manipularDatos("a")
        MsgBox("Se han modificado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
        Limpiar()
    End If
End If
End Sub

Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione " & Ventana & " que desea eliminar",
MsgBoxStyle.Information, Me.Text)
    Else
        Dim Cuantos As Integer = 0
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
            " FROM grupo_producto " & _
            " WHERE gru_pro_codigo LIKE '" & txtCodigo.Text & ".%'
"

        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
        SQL.dataR.Close()
        If Cuantos > 0 Then
            MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de SubGrupos, no se
puede eliminar", MsgBoxStyle.Information, Me.Text)
            Exit Sub
        End If

        Cuantos = 0
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
            " FROM producto " & _
            " WHERE gru_pro_codigo='" & txtCodigo.Text & "' "

        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
        SQL.dataR.Close()
        If Cuantos > 0 Then
            MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionado con datos de Repuestos/Servicios,
no se puede eliminar", MsgBoxStyle.Information, Me.Text)
            Exit Sub
        End If

        If MsgBox("Está seguro que desea eliminar " & Ventana & " " &
txtNombre.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
            manipularDatos("e")

```

```

        MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)
        Limpiar()
    End If
End If
End Sub

Private Sub manipularDatos(ByVal tipo As String)
    If UCase(tipo) = "I" Then
        Dim nivel As Integer
        nivel = CuantosC(1, txtCodigo.Text, ".")
        strSQL = " INSERT INTO
grupo_producto(gru_pro_codigo,gru_pro_nombre,gru_pro_nivel,gru_pro_descripc
ion) VALUES(' " & _
                Trim(txtCodigo.Text) & "','" & _
                Texto(txtNombre.Text) & "','" & _
                nivel & "','" & _
                Texto(txtDescripcion.Text) & "'" )"
        SQL.Ejecutar(strSQL, "L", tipo)
    ElseIf UCase(tipo) = "E" Then
        strSQL = " DELETE FROM grupo_producto " & _
                " WHERE gru_pro_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)
    Else
        strSQL = " UPDATE grupo_producto SET " & _
                " gru_pro_nombre='" & Texto(txtNombre.Text) & "','" & _
                " gru_pro_descripcion='" & Texto(txtDescripcion.Text)
& "'" & _
                " WHERE gru_pro_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)
    End If
End Sub

Private Sub radioSubGrupo_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles radioSubGrupo.CheckedChanged
    comboCodigo.Enabled = True
    comboCodigo.SelectedIndex = -1
    txtCodigo.Text = ""
End Sub

Private Sub cargarCombo()
    strSQL = " SELECT gru_pro_codigo as codigo,gru_pro_nombre as nombre
" & _
            " FROM grupo_producto " & _
            " ORDER BY gru_pro_codigo "
    SQL.Ejecutar(strSQL)
    Dim ds As New DataTable
    ds.Load(SQL.dataR, LoadOption.OverwriteChanges)
    comboCodigo.DisplayMember = ds.Columns.Item(1).Caption
    comboCodigo.ValueMember = ds.Columns.Item(0).Caption
    comboCodigo.DataSource = ds

    ds = Nothing
    SQL.dataR.Close()
    comboCodigo.SelectedIndex = -1
End Sub

```

```

Private Sub radioGrupo_CheckedChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles radioGrupo.CheckedChanged
    comboCodigo.Enabled = False
    comboCodigo.SelectedIndex = -1
    cargarCodigo()
End Sub

Private Sub comboCodigo_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
comboCodigo.SelectedIndexChanged
    If radioSubGrupo.Checked = True And comboCodigo.SelectedValue <> ""
Then
        cargarCodigo()
    End If
End Sub

Private Sub cargarCodigo()
    txtCodigo.Text = ""
    If radioGrupo.Checked = True Then
        strSQL = " SELECT COALESCE(MAX(gru_pro_codigo), '00') " & _
            " FROM grupo_producto " & _
            " WHERE INSTR(gru_pro_codigo, '.')=0 "
        SQL.Ejecutar(strSQL)
        If SQL.dataR.HasRows Then
            SQL.dataR.Read()
            txtCodigo.Text = ((CInt(SQL.dataR.GetValue(0).ToString) +
1).ToString.PadLeft(Len(SQL.dataR.GetValue(0).ToString), "0"))
            SQL.dataR.Close()
        Else
            txtCodigo.Text = "01"
        End If
    Else
        Dim nivel As Integer
        nivel = CuantosC(1, comboCodigo.SelectedValue, ".")
        strSQL = " SELECT COALESCE(MAX(gru_pro_codigo), ' ' & _
            " FROM grupo_producto " & _
            " WHERE gru_pro_codigo LIKE ' ' & _
            " AND gru_pro_nivel=" & nivel + 1
        SQL.Ejecutar(strSQL)
        If SQL.dataR.HasRows Then
            SQL.dataR.Read()
            txtCodigo.Text = codigoSubGrupo(1,
SQL.dataR.GetValue(0).ToString, ".")
            SQL.dataR.Close()
        Else
            txtCodigo.Text = comboCodigo.SelectedValue.ToString & "." &
"01"
        End If
    End If
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click

```



```

Dim excel As New Excel.Application
excel.Visible = True

With excel
    .SheetsInNewWorkbook = 1
    .Workbooks.Add()
    .Worksheets(1).Select()

    Dim i As Integer = 1
    For col = 0 To vsfg.Cols.Count - 1
        .Cells(1, i).value = vsfg.Item(0, col)
        .Cells(1, i).EntireRow.Font.Bold = True
        i += 1
    Next

    'i = 2

    Dim k As Integer = 1
    For col = 0 To vsfg.Cols.Count - 1
        i = 1
        For row = 0 To vsfg.Rows.Count - 1
            .Cells(i, k).Value = vsfg.Item(row, col)
            i += 1
        Next
        k += 1
    Next
End With
End Sub
End Class

```

Lista de Precios

```

Public Class frmListaPrecioxProductos
    Private strSQL As String
    Private SQL As New clsSQL
    Public producto As String
    Private Const Ventana As String = "la Lista de Precio"
    Private Sub frmListaPrecioxProductos_Load(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        If producto <> "" Then
            vsfg.Cols(4).Visible = False
        Else
            vsfg.Cols(4).Visible = True
        End If
        cargarCombo()
        Limpiar()
    End Sub
    Public Sub cargarCombo()
        Dim ds As New DataTable
        strSQL = " SELECT 1,'% ' as codigo,'Todos los productos' as nombre "
& _
            " UNION " & _
            " SELECT 2,pro_codigo as codigo,pro_nombre as nombre " & _
            " FROM producto " & _

```

```

        " ORDER BY 1,nombre "
SQL.Ejecutar(strSQL)
ds.Load(SQL.dataR, LoadOption.OverwriteChanges)
combo.DisplayMember = ds.Columns.Item(2).Caption
combo.ValueMember = ds.Columns.Item(1).Caption
combo.DataSource = ds

ds = Nothing
SQL.dataR.Close()
If producto <> "" Then
    combo.SelectedValue = producto
End If
End Sub

Private Sub Limpiar()
    producto = ""
    vsfg.Focus()
    Cargar()
End Sub

Private Sub Cargar()
    Dim row As Integer, i As Integer
    strSQL = " SELECT pro_codigo,pro_nombre,pro_descripcion,pro_costo"
& _
        " FROM producto " & _
        " WHERE pro_codigo LIKE '" & combo.SelectedValue & "' " & _
        " ORDER BY pro_codigo "
SQL.Ejecutar(strSQL)
vsfg.Rows.Count = 1
row = 1
While SQL.dataR.Read()
    vsfg.Rows.Add()
    For i = 0 To SQL.dataR.FieldCount - 1
        vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
    Next
    row = row + 1
End While
SQL.dataR.Close()

For i = 1 To vsfg.Rows.Count - 1
    vsfg.Item(i, 0) = CStr(i)
Next
End Sub

Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub

Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
    Limpiar()
End Sub

Private Sub vsfg_AfterEdit(ByVal sender As System.Object, ByVal e As
C1.Win.C1FlexGrid.RowColEventArgs) Handles vsfg.AfterEdit

```

```

Dim num As Double

If e.Col = 4 Then
    If IsNumeric(vsfg.Item(e.Row, e.Col)) Then

        vsfg.Item(e.Row, e.Col) =
FormatNumber(Número(vsfg.Item(e.Row, e.Col)), 2)
        If CDBl(vsfg.Item(e.Row, 4)) = 0 Then
            vsfg.Item(e.Row, e.Col) = FormatNumber("0", 2)
        End If
    Else
        vsfg.Item(e.Row, e.Col) = FormatNumber("0", 2)
    End If
End If
num = CDBl(vsfg.Item(e.Row, 4))
strSQL = " UPDATE producto SET " & _
        "pro_nombre=" & vsfg.Item(e.Row, 2) & "," & _
        "pro_descripcion=" & vsfg.Item(e.Row, 3) & "," & _
        "pro_costo=" & NBDD(num) & "" & _
        "WHERE pro_codigo=" & vsfg.Item(e.Row, 1) & ""
SQL.Ejecutar(strSQL, "L", "I")
End Sub

Private Sub vsfg_BeforeEdit(ByVal sender As Object, ByVal e As
C1.Win.C1FlexGrid.RowColEventArgs) Handles vsfg.BeforeEdit
    If e.Col <> 4 Then
        e.Cancel = True
    End If
End Sub

Private Sub combo_SelectedIndexChanged(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles combo.SelectedIndexChanged
    Cargar()
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
            .Cells(1, i).EntireRow.Font.Bold = True
            i += 1
        Next
        Dim k As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            i = 1
            For row = 0 To vsfg.Rows.Count - 1

```

```

        .Cells(i, k).Value = vsfg.Item(row, col)
        i += 1
    Next
    k += 1
Next
End With
End Sub
End Class

```

Listado de Repuestos/Servicios

```

Public Class frmListadoProductos
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "el Cliente"
    Private Sub frmListadoProductos_Load(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        cargarCombos()
        Limpiar()
    End Sub
    Public Sub cargarCombos()
        Dim ds0 As New DataTable, ds1 As New DataTable, ds2 As New
DataTable

        strSQL = " SELECT 1, '%' as codigo, 'Todos los tipos' as nombre " & _
            " UNION " & _
            " SELECT 2, 'R' as codigo, 'REPUESTOS' as nombre " & _
            " UNION " & _
            " SELECT 3, 'S' as codigo, 'SERVICIOS' as nombre " & _
            " ORDER BY 1, nombre "
        SQL.Ejecutar(strSQL)
        ds2.Load(SQL.dataR, LoadOption.OverwriteChanges)
        comboTipo.DisplayMember = ds2.Columns.Item(2).Caption
        comboTipo.ValueMember = ds2.Columns.Item(1).Caption
        comboTipo.DataSource = ds2
        SQL.dataR.Close()

        strSQL = " SELECT 1, '%' as codigo, 'Todos los Grupos' as nombre " &
_
            " UNION " & _
            " SELECT 2, gru_pro_codigo as
codigo, CONCAT(space(gru_pro_nivel*3), cast(gru_pro_nombre as char(30))) as
nombre " & _
            " FROM grupo_producto " & _
            " ORDER BY 1, codigo "
        SQL.Ejecutar(strSQL)
        ds0.Load(SQL.dataR, LoadOption.OverwriteChanges)
        comboGrupo.DisplayMember = ds0.Columns.Item(2).Caption
        comboGrupo.ValueMember = ds0.Columns.Item(1).Caption
        comboGrupo.DataSource = ds0
        SQL.dataR.Close()
    End Sub

    Private Sub Limpiar()

```

```

        txtCodigoFiltro.Text = ""
        txtNombreFiltro.Text = ""
        txtMarca.Text = ""
        comboTipo.SelectedIndex = 0
        comboGrupo.SelectedIndex = 0
        Cargar()
        vsfg.Focus()
    End Sub

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

    Private Sub Cargar()
        Dim row As Integer, i As Integer, col As Integer
        col = 14
        vsfg.Cols.Count = col
        strSQL = " "

        strSQL = " SELECT
pro_tipo,pro_codigo,pro_nombre,gru_pro_nombre,pro_marca,pro_costo,pro_ubica
cion,pro_max,pro_min,pro_pto_pedido,pro_num_mecanico,pro_tiempo,pro_descrip
cion " & _
            " FROM producto " & _
            " LEFT JOIN grupo_producto " & _
            " ON producto.gru_pro_codigo=grupo_producto.gru_pro_codigo
" & _
            " WHERE pro_nombre LIKE '%" & Trim(txtNombreFiltro.Text) &
"%" " & _
            " AND pro_codigo LIKE '%" & Trim(txtCodigoFiltro.Text) &
"%" " & _
            " AND pro_tipo LIKE '%" & comboTipo.SelectedValue.ToString
& "%" "
        strSQL = strSQL & " AND producto.pro_marca LIKE '%" &
Trim(txtMarca.Text) & "%" " & _
            " AND producto.gru_pro_codigo LIKE '" &
comboGrupo.SelectedValue.ToString & "' " & _
            " ORDER BY pro_tipo,pro_nombre "
        SQL.Ejecutar(strSQL)
        vsfg.Rows.Count = 1
        row = 1
        While SQL.dataR.Read()
            vsfg.Rows.Add()
            For i = 0 To SQL.dataR.FieldCount - 1
                vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
            Next
            row = row + 1
        End While
        SQL.dataR.Close()

        For i = 1 To vsfg.Rows.Count - 1
            vsfg.Item(i, 0) = CStr(i)
        Next

    End Sub

```

```

    Private Sub btnBuscar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnBuscar.Click
        Cargar()
    End Sub

    Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
        Dim excel As New Excel.Application
        excel.Visible = True

        With excel
            .SheetsInNewWorkbook = 1
            .Workbooks.Add()
            .Worksheets(1).Select()

            Dim i As Integer = 1
            For col = 0 To vsfg.Cols.Count - 1
                .Cells(1, i).value = vsfg.Item(0, col)
                .Cells(1, i).EntireRow.Font.Bold = True
                i += 1
            Next
            Dim k As Integer = 1
            For col = 0 To vsfg.Cols.Count - 1
                i = 1
                For row = 0 To vsfg.Rows.Count - 1
                    .Cells(i, k).Value = vsfg.Item(row, col)
                    i += 1
                Next
                k += 1
            Next
        End With
    End Sub
End Class

```

Existencia de Inventarios

```

Public Class frmExistencias
    Private strSQL As String
    Private SQL As New clsSQL
    Public producto As String
    Private Const Ventana As String = "la Existencia de Repuesto"
    Private Sub frmExistencias_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        Limpiar()
    End Sub
    Private Sub Limpiar()
        vsfg.Focus()
        Cargar()
        txtInv.Text = ""
        txtPtoServMAX.Text = ""
        txtPtoServMIN.Text = ""
    End Sub

```

```

Private Sub Cargar()
    Dim row As Integer, i As Integer
    strSQL = " SELECT
existencia.pro_codigo,pro_nombre,existencia.exi_cantidad " & _
            " FROM existencia " & _
            " INNER JOIN producto " & _
            " ON producto.pro_codigo=existencia.pro_codigo " & _
            " ORDER BY pro_nombre "
    SQL.Ejecutar(strSQL)
    vsfg.Rows.Count = 1
    row = 1
    While SQL.dataR.Read()
        vsfg.Rows.Add()
        For i = 0 To SQL.dataR.FieldCount - 1
            vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
        Next
        row = row + 1
    End While
    SQL.dataR.Close()

    For i = 1 To vsfg.Rows.Count - 1
        vsfg.Item(i, 0) = CStr(i)
    Next
End Sub

Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub

Private Sub vsfg_AfterEdit(ByVal sender As System.Object, ByVal e As
C1.Win.C1FlexGrid.RowColEventArgs) Handles vsfg.AfterEdit
    If e.Col = 3 Then

        strSQL = " SELECT producto.pro_max, producto.pro_min" & _
            " FROM producto " & _
            " INNER JOIN existencia " & _
            " ON producto.pro_codigo=existencia.pro_codigo " & _
            " WHERE existencia.pro_codigo='" & vsfg.Item(e.Row, 1) & "'
" & _
            " AND producto.pro_tipo='R' "

        SQL.Ejecutar(strSQL)
        If SQL.dataR.HasRows Then
            SQL.dataR.Read()
            txtPtoServMAX.Text = SQL.dataR.GetValue(0).ToString
            txtPtoServMIN.Text = SQL.dataR.GetValue(1).ToString
            SQL.dataR.Close()
        End If

        txtInv.Text = vsfg.Item(e.Row, 3)

        If IsNumeric(vsfg.Item(e.Row, e.Col)) Then

```

```

                vsfg.Item(e.Row, e.Col) = FormatNumber(vsfg.Item(e.Row,
e.Col), 0)
            Else
                vsfg.Item(e.Row, e.Col) = FormatNumber("0", 0)
            End If
        End If
        If CInt(txtInv.Text) <= CInt(txtPtoServMAX.Text) And
CInt(txtInv.Text) >= CInt(txtPtoServMIN.Text) Then

            If CInt(txtInv.Text) <= CInt(txtPtoServMIN.Text + 1) Then
                strSQL = "UPDATE producto set pro_pto_pedido='1' WHERE
pro_codigo='" & vsfg.Item(e.Row, 1) & "' & _
                    "AND pro_tipo='R'"
                SQL.Ejecutar(strSQL)
                SQL.dataR.Close()
            Else
                strSQL = "UPDATE producto set pro_pto_pedido='0' WHERE
pro_codigo='" & vsfg.Item(e.Row, 1) & "' & _
                    "AND pro_tipo='R'"
                SQL.Ejecutar(strSQL)
                SQL.dataR.Close()
            End If
            strSQL = " REPLACE INTO existencia(pro_codigo,exi_cantidad)
VALUES('" & _
                vsfg.Item(e.Row, 1) & "','" & _
                vsfg.Item(e.Row, 3) & "')"
            SQL.Ejecutar(strSQL, "L", "I")
        ElseIf CInt(txtInv.Text) > CInt(txtPtoServMAX.Text) Then
            MsgBox("El valor máximo permitido es " & txtPtoServMAX.Text & "
", MsgBoxStyle.Information, Me.Text)
        Else
            MsgBox("El valor mínimo permitido es " & txtPtoServMIN.Text & "
", MsgBoxStyle.Information, Me.Text)
        End If
        Limpiar()

    End Sub

    Private Sub vsfg_BeforeEdit(ByVal sender As Object, ByVal e As
C1.Win.C1FlexGrid.RowColEventArgs) Handles vsfg.BeforeEdit
        If e.Col <> 3 Then
            e.Cancel = True
        End If
    End Sub

    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
        Limpiar()
    End Sub

    Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
        Dim excel As New Excel.Application
        excel.Visible = True

```



```

With excel
    .SheetsInNewWorkbook = 1
    .Workbooks.Add()
    .Worksheets(1).Select()

    Dim i As Integer = 1
    For col = 0 To vsfg.Cols.Count - 1
        .Cells(1, i).value = vsfg.Item(0, col)
        .Cells(1, i).EntireRow.Font.Bold = True
        i += 1
    Next

    'i = 2

    Dim k As Integer = 1
    For col = 0 To vsfg.Cols.Count - 1
        i = 1
        For row = 0 To vsfg.Rows.Count - 1
            .Cells(i, k).Value = vsfg.Item(row, col)
            i += 1
        Next
        k += 1
    Next
End With
End Sub
End Class

```

Aseguradoras

```

Public Class frmAseguradoras
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "la Aseguradora"
    Private htCiudades As New Hashtable
    Private Sub frmAseguradoras_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        cargarCombos()
        Limpiar()
    End Sub
    Private Sub Limpiar()
        txtNombre.Text = ""
        txtRuc.Text = ""
        txtDireccion.Text = ""
        txtTelefono.Text = ""
        txtCelular.Text = ""
        txtCiudad.Text = ""
        txtFax.Text = ""
        txtEMail.Text = ""
        txtWeb.Text = ""
        txtObservacion.Text = ""
        btnContacto.Enabled = False
        btnAceptar.Text = "&Aceptar"
        btnEliminar.Enabled = False
        txtNombre.Focus()
    End Sub

```

```

        Cargar()
    End Sub

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
        Limpiar()
    End Sub

    Private Sub Cargar()
        Dim row As Integer, i As Integer
        strSQL = " SELECT asg_codigo,asg_nombre,asg_ruc, " & _
            "
asg_direccion,asg_telefono,asg_fax,asg_celular,ase_ciudad,asg_email,asg_web
,asg_observaciones " & _
            " FROM aseguradora " & _
            " ORDER BY asg_nombre "
        SQL.Ejecutar(strSQL)
        vsfg.Rows.Count = 1
        row = 1
        While SQL.dataR.Read()
            vsfg.Rows.Add()
            For i = 0 To SQL.dataR.FieldCount - 1
                vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
            Next
            row = row + 1
        End While
        SQL.dataR.Close()

        For i = 1 To vsfg.Rows.Count - 1
            vsfg.Item(i, 0) = CStr(i)
            vsfg.Item(i, 8) = vsfg.Item(i, 8)
        Next
    End Sub

    Private Function controlDatos() As Boolean
        If Trim(txtNombre.Text) = "" Then
            MsgBox("Ingreso el nombre para " & Ventana,
MsgBoxStyle.Information, Me.Text)
            txtNombre.Focus()
            Return False
        ElseIf Trim(txtRuc.Text) = "" Then
            MsgBox("Ingreso el ci/ruc para " & Ventana,
MsgBoxStyle.Information, Me.Text)
            txtRuc.Focus()
            Return False
        Else
            Return True
        End If
    End Function

```

```

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If btnAceptar.Text = "&Aceptar" Then
        If controlDatos() = True Then
            If VerificarID(txtRuc.Text) Then
                manipularDatos("i")
                MsgBox("Se han ingresado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
                Limpiar()
            End If
        End If
    Else
        If controlDatos() = True Then
            If VerificarID(txtRuc.Text) Then
                manipularDatos("a")
                MsgBox("Se han modificado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
                Limpiar()
            End If
        End If
    End If
End Sub

```

```

Private Sub datosAseguradora()
    Dim row As Integer
    row = vsfg.Row

    txtCodigo.Text = vsfg.Item(row, 1)
    txtNombre.Text = vsfg.Item(row, 2)
    txtRuc.Text = vsfg.Item(row, 3)
    txtDireccion.Text = vsfg.Item(row, 4)
    txtTelefono.Text = vsfg.Item(row, 5)
    txtFax.Text = vsfg.Item(row, 6)
    txtCelular.Text = vsfg.Item(row, 7)
    txtCiudad.Text = vsfg.Item(row, 8)
    txtEMail.Text = vsfg.Item(row, 9)
    txtWeb.Text = vsfg.Item(row, 10)
    txtObservacion.Text = vsfg.Item(row, 11)

    btnAceptar.Text = "&Modificar"
    txtCodigo.Enabled = False
    btnEliminar.Enabled = True
    btnContacto.Enabled = True
End Sub

```

```

Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione " & Ventana & " que desea eliminar",
MsgBoxStyle.Information, Me.Text)
    Else
        Dim Cuantos As Integer = 0
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
            " FROM proforma " & _
            " WHERE asg_codigo='" & txtCodigo.Text & "'"
    End If

```

```

SQL.Ejecutar(strSQL)
SQL.dataR.Read()
Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
SQL.dataR.Close()
If Cuantos > 0 Then
    MsgBox(Ventana.Remove(1, Ventana.Length - 1).ToUpper &
Ventana.Remove(0, 1) & " está relacionada con datos de Proformas, no se
puede eliminar", MsgBoxStyle.Information, Me.Text)
    Exit Sub
End If

If MsgBox("Está seguro que desea eliminar " & Ventana & " " &
txtNombre.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
    manipularDatos("e")
    MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)
    Limpiar()
End If
End If
End Sub

Private Sub manipularDatos(ByVal tipo As String)
If UCase(tipo) = "I" Then
    strSQL = " INSERT INTO aseguradora(asg_ruc," & _
"asg_nombre,asg_direccion,asg_telefono,asg_celular,asg_fax," & _
"ase_ciudad,asg_email,asg_web,asg_observaciones)
VALUES(' " & _
Trim(txtRuc.Text) & "','" & _
Texto(txtNombre.Text) & "','" & _
Trim(txtDireccion.Text) & "','" & _
Trim(txtTelefono.Text) & "','" & _
Trim(txtCelular.Text) & "','" & _
Trim(txtFax.Text) & "','" & _
Trim(txtCiudad.Text) & "','" & _
Trim(txtEMail.Text) & "','" & _
Trim(txtWeb.Text) & "','" & _
Texto(txtObservacion.Text) & "')"
SQL.Ejecutar(strSQL, "L", tipo)
ElseIf UCase(tipo) = "E" Then
    strSQL = " DELETE FROM aseguradora " & _
" WHERE asg_codigo='" & txtCodigo.Text & "' "
SQL.Ejecutar(strSQL, "L", tipo)
Else
    strSQL = " UPDATE aseguradora SET " & _
" asg_ruc='" & Trim(txtRuc.Text) & "','" & _
" asg_nombre='" & Texto(txtNombre.Text) & "','" & _
" asg_direccion='" & Trim(txtDireccion.Text) & "','" & _
" asg_telefono='" & Trim(txtTelefono.Text) & "','" & _
" asg_celular='" & Trim(txtCelular.Text) & "','" & _
" asg_fax='" & Trim(txtFax.Text) & "','" & _
" ase_ciudad='" & Trim(txtCiudad.Text) & "','" & _
" asg_email='" & Trim(txtEMail.Text) & "','" & _
" asg_web='" & Trim(txtWeb.Text) & "','" & _

```

```

        " asg_observaciones='" & Texto(txtObservacion.Text) &
''' & _
        " WHERE asg_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)
    End If
End Sub

Private Sub btnContacto_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnContacto.Click
    If frmContactosA.Visible Then
        frmContactosA.Activate()
    Else
        frmContactosA.MdiParent = mdiPrincipal
        frmContactosA.Persona = txtCodigo.Text
        frmContactosA.Show()
    End If
End Sub

Private Sub vsfg_DoubleClick(ByVal sender As Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
    datosAseguradora()
End Sub

Private Sub txtTelefono_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtTelefono.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub txtCelular_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtCelular.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub txtFax_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtFax.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub txtRuc_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtRuc.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()
    End With
End Sub

```

```

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            .Cells(1, i).value = vsfg.Item(0, col)
            .Cells(1, i).EntireRow.Font.Bold = True
            i += 1
        Next

        Dim k As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1
            i = 1
            For row = 0 To vsfg.Rows.Count - 1
                .Cells(i, k).Value = vsfg.Item(row, col)
                i += 1
            Next
            k += 1
        Next
    End With
End Sub
End Class

```

Asesores

```

Public Class frmAsesores
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "el Asesor"
    Private Sub frmAsesores_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        Limpiar()
    End Sub
    Private Sub Limpiar()
        txtNombre.Text = ""
        txtApellido.Text = ""
        txtRuc.Text = ""
        txtCodigoAuxiliar.Text = ""
        txtTelefono.Text = ""
        txtCelular.Text = ""
        txtEMail.Text = ""
        txtCodigo.Text = ""
        txtActualiza.Text = ""
        txtIngresa.Text = ""
        btnAceptar.Text = "&Aceptar"
        btnEliminar.Enabled = False
        Cargar()
        txtNombre.Focus()
    End Sub

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

```

```

Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
    Limpiar()
End Sub

Private Sub Cargar()
    Dim row As Integer, i As Integer
    strSQL = " SELECT
ase_codigo,ase_nombre,ase_apellido,ase_ruc,ase_telefono,ase_celular,ase_ema
il,ase_codigo_auxiliar " & _
        " FROM asesor " & _
        " ORDER BY ase_apellido,ase_nombre "
    SQL.Ejecutar(strSQL)
    vsfg.Rows.Count = 1
    row = 1
    While SQL.dataR.Read()
        vsfg.Rows.Add()
        For i = 0 To SQL.dataR.FieldCount - 1
            vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
        Next
        row = row + 1
    End While
    SQL.dataR.Close()

    For i = 1 To vsfg.Rows.Count - 1
        vsfg.Item(i, 0) = CStr(i)
    Next
End Sub

Private Function controlDatos() As Boolean
    If Trim(txtNombre.Text) = "" Then
        MsgBox("Ingrese el nombre para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtNombre.Focus()
    ElseIf Trim(txtApellido.Text) = "" Then
        MsgBox("Ingrese el apellido para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtApellido.Focus()
    Return False
    Else
        Return True
    End If
End Function

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    txtIngresa.Text = "i"
    txtActualiza.Text = "a"

    strSQL = " SELECT COALESCE(COUNT(*),0) " & _
        " FROM asesor " & _
        " WHERE ase_ruc='" & Trim(txtRuc.Text) & "' "
    SQL.Ejecutar(strSQL)

    If btnAceptar.Text = "&Aceptar" Then
        If controlDatos() = True Then

```

```

        If VerificarID(txtRuc.Text) Then
            If SQL.dataR.HasRows Then
                MsgBox("La CI/RUC ya se encuentra ingresada!",
vbOKOnly + vbCritical, "Error")
                SQL.dataR.Close()
                If MsgBox("Este CI/RUC ya está registrado, desea
cargar la información actual?", MsgBoxStyle.Question + MsgBoxStyle.YesNo,
Me.Text) = MsgBoxResult.Yes Then
                    CargarRuc(Trim(txtRuc.Text))
                    txtActualiza.Text = "a"
                    txtIngresa.Text = "i"
                Else
                    txtRuc.Text = ""
                    txtRuc.Focus()
                End If
            Else
                SQL.dataR.Close()
                manipularDatos("i")
                MsgBox("Se han ingresado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
                Limpiar()
            End If
        End If
    End If
    txtActualiza.Text = ""
    txtIngresa.Text = ""
    SQL.dataR.Close()
Else
    If controlDatos() = True Then
        If VerificarID(txtRuc.Text) Then
            SQL.dataR.Close()
            manipularDatos("a")
            MsgBox("Se han modificado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    End If
End If
End Sub

Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione " & Ventana & " que desea eliminar",
MsgBoxStyle.Information, Me.Text)
    Else
        Dim Cuantos As Integer = 0, entro As Boolean = False
        Cuantos = 0
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
                " FROM persona " & _
                " WHERE ase_nombre='" & txtNombre.Text & "' " & _
                " AND per_que='C' "
        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
        SQL.dataR.Close()
    End If
End Sub

```



```

        If Cuantos > 0 Then
            If MsgBox("Este Asesor está asignado a datos de Clientes,
desea continuar?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.No Then
                entro = True
                Exit Sub
            Else
                manipularDatos("e")
                MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)
                Limpiar()
                Exit Sub
            End If
        End If

        If MsgBox("Está seguro que desea eliminar " & Ventana & " " &
txtApellido.Text & " " & txtNombre.Text & "?", MsgBoxStyle.Question +
MsgBoxStyle.YesNo, Me.Text) = MsgBoxResult.Yes Then
            manipularDatos("e")
            MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    End If
End Sub

Private Sub manipularDatos(ByVal tipo As String)
    If UCase(tipo) = "I" Then
        strSQL = " INSERT INTO asesor(ase_nombre,ase_apellido,ase_ruc,"
& _
"ase_codigo_auxiliar,ase_telefono,ase_celular,ase_email) VALUES(' " & _
        Texto(txtNombre.Text) & "','" & _
        Texto(txtApellido.Text) & "','" & _
        Trim(txtRuc.Text) & "','" & _
        Texto(txtCodigoAuxiliar.Text) & "','" & _
        Trim(txtTelefono.Text) & "','" & _
        Trim(txtCelular.Text) & "','" & _
        Trim(txtEMail.Text) & "'"
        SQL.Ejecutar(strSQL, "L", tipo)
    ElseIf UCase(tipo) = "E" Then
        strSQL = " DELETE FROM asesor " & _
            " WHERE ase_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)
        strSQL = " UPDATE persona SET " & _
            " ase_nombre='" & _
            " WHERE per_que='C' "
        SQL.Ejecutar(strSQL, "L", tipo)
    Else
        strSQL = " UPDATE asesor SET " & _
            " ase_nombre='" & Texto(txtNombre.Text) & "','" & _
            " ase_apellido='" & Texto(txtApellido.Text) & "','" & _
            " ase_ruc='" & Trim(txtRuc.Text) & "','" & _
            " ase_codigo_auxiliar='" &
Texto(txtCodigoAuxiliar.Text) & "','" & _
            " ase_telefono='" & Trim(txtTelefono.Text) & "','" & _

```

```

        " ase_celular='" & Trim(txtCelular.Text) & "'," & _
        " ase_email='" & Trim(txtEMail.Text) & "'" & _
        " WHERE ase_codigo='" & txtCodigo.Text & "'"
    SQL.Ejecutar(strSQL, "L", tipo)
End If
End Sub

Private Sub vsfg_DoubleClick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
    datosAsesor()
End Sub

Private Sub datosAsesor()
    Dim row As Integer
    row = vsfg.Row

    txtCodigo.Text = vsfg.Item(row, 1)
    txtNombre.Text = vsfg.Item(row, 2)
    txtApellido.Text = vsfg.Item(row, 3)
    txtRuc.Text = vsfg.Item(row, 4)
    txtCodigoAuxiliar.Text = vsfg.Item(row, 8)
    txtTelefono.Text = vsfg.Item(row, 5)
    txtCelular.Text = vsfg.Item(row, 6)
    txtEMail.Text = vsfg.Item(row, 7)
    btnAceptar.Text = "&Modificar"
    btnEliminar.Enabled = True
End Sub

Private Sub txtRuc_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtRuc.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub txtRuc_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtRuc.LostFocus
    If (txtActualiza.Text <> "a" Or txtIngresa.Text <> "i") Then
        Dim Cuantos As Integer = 0
        strSQL = " SELECT COALESCE(COUNT(*),0) " & _
            " FROM asesor " & _
            " WHERE ase_ruc='" & Trim(txtRuc.Text) & "'"
        SQL.Ejecutar(strSQL)
        If SQL.dataR.HasRows Then
            Cuantos = CInt(SQL.dataR.GetValue(0).ToString)
        End If
        SQL.dataR.Close()
        If Cuantos > 0 Then
            If MsgBox("Este CI/RUC ya está registrado, desea cargar la
información actual?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
                CargarRuc(Trim(txtRuc.Text))
            Else
                txtRuc.Text = ""
                txtRuc.Focus()
            End If
        End If
    End If
End Sub

```

```

End If

End Sub
Private Sub CargarRuc(ByVal txt1 As String)
    Dim row As Integer
    row = vsfg.Row
    Limpiar()
    strSQL = "SELECT
ase_codigo,ase_nombre,ase_apellido,ase_ruc,ase_telefono,ase_celular,ase_ema
il,ase_codigo_auxiliar " & _
            " FROM asesor " & _
            "WHERE ase_ruc='" & txt1 & "'"

    SQL.Ejecutar(strSQL)
    If SQL.dataR.HasRows Then
        SQL.dataR.Read()
        txtNombre.Text = SQL.dataR.GetValue(1)
        txtApellido.Text = Valor(SQL.dataR, 2)
        txtRuc.Text = SQL.dataR.GetValue(3)
        txtTelefono.Text = Valor(SQL.dataR, 4)
        txtCelular.Text = Valor(SQL.dataR, 5)
        txtEMail.Text = Valor(SQL.dataR, 6)
        txtCodigoAuxiliar.Text = Valor(SQL.dataR, 7)
        btnAceptar.Text = "&Modificar"
        btnEliminar.Enabled = True
        txtCodigo.Text = SQL.dataR.GetValue(0)
        SQL.dataR.Close()
    End If
    SQL.dataR.Close()
End Sub
Private Sub txtTelefono_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtTelefono.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub
Private Sub txtCelular_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtCelular.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub
Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

    With excel
        .SheetsInNewWorkbook = 1
        .Workbooks.Add()
        .Worksheets(1).Select()

        Dim i As Integer = 1
        For col = 0 To vsfg.Cols.Count - 1

```

```

        .Cells(1, i).value = vsfg.Item(0, col)
        .Cells(1, i).EntireRow.Font.Bold = True
        i += 1
    Next
    Dim k As Integer = 1
    For col = 0 To vsfg.Cols.Count - 1
        i = 1
        For row = 0 To vsfg.Rows.Count - 1
            .Cells(i, k).Value = vsfg.Item(row, col)
            i += 1
        Next
        k += 1
    Next
End With
End Sub
End Class

```

Formas de Pago

```

Public Class frmFormasPago
    Private strSQL As String
    Private SQL As New clsSQL
    Private Const Ventana As String = "la Forma de Pago"
    Private Sub frmFormasPago_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        Limpiar()
    End Sub
    Private Sub Limpiar()
        txtCodigo.Text = ""
        txtNombre.Text = ""
        txtDescripcion.Text = ""
        txtTiempo.Text = ""
        btnAceptar.Text = "&Aceptar"
        txtNombre.Focus()
        btnEliminar.Enabled = False
        cargar()
    End Sub

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
        Limpiar()
    End Sub

    Private Sub Cargar()
        Dim row As Integer, i As Integer
        strSQL = " SELECT
for_pag_codigo,for_pag_nombre,for_pag_descripcion,for_pag_tiempo " & _
        " FROM forma_pago " & _
        " ORDER BY for_pag_nombre "
    End Sub

```

```

SQL.Ejecutar(strSQL)
vsfg.Rows.Count = 1
row = 1
While SQL.dataR.Read()
    vsfg.Rows.Add()
    For i = 0 To SQL.dataR.FieldCount - 1
        vsfg.Item(row, i + 1) = Valor(SQL.dataR, i)
    Next
    row = row + 1
End While
SQL.dataR.Close()

For i = 1 To vsfg.Rows.Count - 1
    vsfg.Item(i, 0) = CStr(i)
Next
End Sub

Private Sub datosColor()
    Dim row As Integer
    row = vsfg.Row

    txtCodigo.Text = vsfg.Item(row, 1)
    txtNombre.Text = vsfg.Item(row, 2)
    txtDescripcion.Text = vsfg.Item(row, 3)
    txtTiempo.Text = vsfg.Item(row, 4)

    btnAceptar.Text = "&Modificar"
    txtCodigo.Enabled = False
    btnEliminar.Enabled = True
End Sub

Private Sub vsfg_DoubleClick(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
    datosColor()
End Sub

Private Function controlDatos() As Boolean
    If Trim(txtNombre.Text) = "" Then
        MsgBox("Ingrese un nombre para " & Ventana,
MsgBoxStyle.Information, Me.Text)
        txtNombre.Focus()
        Return False
    Else
        Return True
    End If
End Function

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If btnAceptar.Text = "&Aceptar" Then
        If controlDatos() = True Then
            manipularDatos("i")
            MsgBox("Se han ingresado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    End If

```

```

Else
    If controlDatos() = True Then
        manipularDatos("a")
        MsgBox("Se han modificado los datos correctamente",
MsgBoxStyle.Information, Me.Text)
        Limpiar()
    End If
End If
End Sub

Private Sub btnEliminar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnEliminar.Click
    If txtCodigo.Text = "" Then
        MsgBox("Seleccione " & Ventana & " que desea eliminar",
MsgBoxStyle.Information, Me.Text)
    Else
        Dim Cuantos As Integer = 0
        If MsgBox("Está seguro que desea eliminar " & Ventana & " " &
txtNombre.Text & "?", MsgBoxStyle.Question + MsgBoxStyle.YesNo, Me.Text) =
MsgBoxResult.Yes Then
            manipularDatos("e")
            MsgBox("Se ha eliminado " & Ventana & " correctamente",
MsgBoxStyle.Information, Me.Text)
            Limpiar()
        End If
    End If
End Sub

Private Sub manipularDatos(ByVal tipo As String)
    If UCase(tipo) = "I" Then
        strSQL = " INSERT INTO
forma_pago(for_pag_nombre,for_pag_descripcion,for_pag_tiempo) VALUES(' & _
                Texto(txtNombre.Text) & ',' & _
                Trim(txtDescripcion.Text) & ',' & _
                NBDD(txtTiempo.Text) & '' )"
        SQL.Ejecutar(strSQL, "L", tipo)
    ElseIf UCase(tipo) = "E" Then
        strSQL = " DELETE FROM forma_pago " & _
                " WHERE for_pag_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)
    Else
        strSQL = " UPDATE forma_pago SET " & _
                " for_pag_nombre='" & Texto(txtNombre.Text) & ',' & _
                " for_pag_descripcion='" & Trim(txtDescripcion.Text) &
                "' & _
                " for_pag_tiempo='" & NBDD(txtTiempo.Text) & "' & _
                " WHERE for_pag_codigo='" & txtCodigo.Text & "' "
        SQL.Ejecutar(strSQL, "L", tipo)
    End If
End Sub

Private Sub btnExcel_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnExcel.Click
    Dim excel As New Excel.Application
    excel.Visible = True

```

```

With excel
    .SheetsInNewWorkbook = 1
    .Workbooks.Add()
    .Worksheets(1).Select()

    Dim i As Integer = 1
    For col = 0 To vsfg.Cols.Count - 1
        .Cells(1, i).value = vsfg.Item(0, col)
        .Cells(1, i).EntireRow.Font.Bold = True
        i += 1
    Next
    Dim k As Integer = 1
    For col = 0 To vsfg.Cols.Count - 1
        i = 1
        For row = 0 To vsfg.Rows.Count - 1
            .Cells(i, k).Value = vsfg.Item(row, col)
            i += 1
        Next
        k += 1
    Next
End With
End Sub
End Class

```

Proforma

```

Public Class frmProforma
    Private strSQL As String
    Private SQL As New clsSQL
    Private SQLaux As New clsSQL
    Private ht As New Hashtable
    Private htCategoria As New Hashtable
    Private htMarcas As New Hashtable
    Private htAseguradoras As New Hashtable
    Private Cargado As Boolean
    Private htProductos As New Hashtable
    Private htGruposProductos As New Hashtable
    Private htGrupos As New Hashtable
    Private cmb As New ComboBox
    Private cmbG As New ComboBox
    Private Aux As String = "ABCDEFGHIJKLMNÑOPQRSTUVWXYZ"
    Private Sub frmProforma_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        SQLaux.Inicializar(ConnMaster)
        Cargado = False

        strSQL = " SELECT tip_prf_codigo as codigo,tip_prf_nombre as nombre
" & _
                " FROM tipo_proforma " & _
                " ORDER BY tip_prf_codigo "
        SQL.Ejecutar(strSQL)
        Dim extra As New Specialized.ListDictionary

        While SQL.dataR.Read

```

```

        extra.Add(SQL.dataR.GetValue(0), SQL.dataR.GetValue(1))
    End While
    SQL.dataR.Close()
    vsfgTipo.Cols(0).DataMap = extra
    vsfgTipo.Cols(0).Item(0) = 1

    cargarCombo()
    Limpiar()
    Cargado = True
End Sub
Private Sub cargarProductos()
    strSQL = " SELECT pro_codigo as codigo,pro_nombre as nombre,
pro_costo as costo " & _
        " FROM producto " & _
        " ORDER BY pro_nombre ASC "
    SQL.Ejecutar(strSQL)

    Dim ds0 As New DataTable

    cmb.AutoCompleteMode = System.Windows.Forms.AutoCompleteMode.Append
    cmb.AutoCompleteSource = AutoCompleteSource.ListItems
    cmb.DropDownStyle = ComboBoxStyle.DropDown
    cmb.FormattingEnabled = True
    cmb.Name = "cmb"

    ds0.Load(SQL.dataR, LoadOption.OverwriteChanges)
    cmb.DisplayMember = ds0.Columns.Item(1).Caption
    cmb.ValueMember = ds0.Columns.Item(0).Caption
    cmb.DataSource = ds0
    SQL.dataR.Close()
    vsfgDetalle.Cols(1).Editor = cmb
    strSQL = " SELECT gru_pro_codigo as codigo,gru_pro_nombre as nombre
" & _
        " FROM grupo_producto " & _
        " ORDER BY gru_pro_nombre ASC "
    SQL.Ejecutar(strSQL)

    Dim dsx As New DataTable

    cmbG.AutoCompleteSource = AutoCompleteSource.ListItems
    cmbG.DropDownStyle = ComboBoxStyle.DropDownList
    cmbG.Name = "cmbG"

    dsx.Load(SQL.dataR, LoadOption.OverwriteChanges)
    cmbG.DisplayMember = dsx.Columns.Item(1).Caption
    cmbG.ValueMember = dsx.Columns.Item(0).Caption
    cmbG.DataSource = dsx
    SQL.dataR.Close()
    vsfgDetalle.Cols(2).Editor = cmbG
End Sub

Private Sub cargarCombo()
    Dim ds0 As New DataTable, ds1 As New DataTable
    Dim ds2 As New DataTable, ds3 As New DataTable
    Dim ds4 As New DataTable, ds5 As New DataTable

```



```
Dim ds6 As New DataTable, ds7 As New DataTable, ds8 As New
DataTable, ds9 As New DataTable
```

```
strSQL = " SELECT 'N' as codigo,'Natural' as nombre " & _
        " UNION " & _
        " SELECT 'J' as codigo,'Jurídica' as nombre "
SQL.Ejecutar(strSQL)
ds6.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboTipo.DisplayMember = ds6.Columns.Item(1).Caption
comboTipo.ValueMember = ds6.Columns.Item(0).Caption
comboTipo.DataSource = ds6
SQL.dataR.Close()

strSQL = " SELECT 1,'' as codigo,'' as nombre,'' as per_apellido,''
as per_nombre " & _
        " UNION " & _
        " SELECT 2,cast(per_codigo as char)as
codigo,TRIM(CONCAT(per_apellido,' ',per_nombre)) as
nombre,per_apellido,per_nombre " & _
        " FROM persona " & _
        " WHERE per_que='C' " & _
        " ORDER BY 1,per_apellido,per_nombre "
SQL.Ejecutar(strSQL)
ds0.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboCliente.DisplayMember = ds0.Columns.Item(2).Caption
comboCliente.ValueMember = ds0.Columns.Item(1).Caption
comboCliente.DataSource = ds0
txtContacto.Text = comboCliente.SelectedValue
SQL.dataR.Close()

strSQL = " SELECT 1,'' as codigo,'' as nombre,'' as per_apellido,''
as per_nombre " & _
        " UNION " & _
        " SELECT 2,cast(per_codigo as char) as
codigo,TRIM(CONCAT(per_apellido,' ',per_nombre)) as
nombre,per_apellido,per_nombre " & _
        " FROM persona " & _
        " WHERE per_que='C' " & _
        " ORDER BY 1,per_apellido,per_nombre "
SQL.Ejecutar(strSQL)
ds8.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboAsegurado.DisplayMember = ds8.Columns.Item(2).Caption
comboAsegurado.ValueMember = ds8.Columns.Item(1).Caption
comboAsegurado.DataSource = ds8
SQL.dataR.Close()

strSQL = " SELECT 1,'' as codigo,'' as nombre " & _
        " UNION " & _
        " SELECT 2,cast(asg_codigo as char) as codigo,asg_nombre
as nombre " & _
        " FROM aseguradora " & _
        " ORDER BY 1,nombre "
SQL.Ejecutar(strSQL)
ds1.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboAseguradora.DisplayMember = ds1.Columns.Item(2).Caption
comboAseguradora.ValueMember = ds1.Columns.Item(1).Caption
```

```

comboAseguradora.DataSource = ds1
SQL.dataR.Close()

strSQL = " SELECT 1,'% ' as codigo,'Seleccione una...' as nombre " &
-
      " UNION " & _
      " SELECT 2, cast(mar_codigo as char) as codigo,mar_nombre as
nombre " & _
      " FROM marca " & _
      " ORDER BY 1,nombre "
SQL.Ejecutar(strSQL)
ds2.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboMarca.DisplayMember = ds2.Columns.Item(2).Caption
comboMarca.ValueMember = ds2.Columns.Item(1).Caption
comboMarca.DataSource = ds2
SQL.dataR.Close()

strSQL = " SELECT 1,'% ' as codigo,'Seleccione uno...' as nombre " &
-
      " UNION " & _
      " SELECT 2,cast(mod_codigo as char) as codigo,mod_nombre
as nombre " & _
      " FROM modelo " & _
      " ORDER BY 1,nombre "
SQL.Ejecutar(strSQL)
ds3.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboModelo.DisplayMember = ds3.Columns.Item(2).Caption
comboModelo.ValueMember = ds3.Columns.Item(1).Caption
comboModelo.DataSource = ds3
SQL.dataR.Close()

strSQL = " SELECT 1,'' as codigo,'' as nombre " & _
      " UNION " & _
      " SELECT 2,cast(col_codigo as char) as codigo,col_nombre
as nombre " & _
      " FROM color " & _
      " ORDER BY 1,nombre "
SQL.Ejecutar(strSQL)
ds4.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboColor.DisplayMember = ds4.Columns.Item(2).Caption
comboColor.ValueMember = ds4.Columns.Item(1).Caption
comboColor.DataSource = ds4
SQL.dataR.Close()

strSQL = " SELECT 1,'' as codigo,'' as nombre " & _
      " UNION " & _
      " SELECT 2,cast(con_codigo as char) as
codigo,TRIM(CONCAT(con_nombre,' ',con_apellido)) as nombre " & _
      " FROM contacto_aseguradora " & _
      " WHERE asg_codigo=" &
comboAseguradora.SelectedValue & "' " & _
      " ORDER BY 1,nombre "
SQL.Ejecutar(strSQL)
ds7.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboReferencia.DisplayMember = ds7.Columns.Item(2).Caption
comboReferencia.ValueMember = ds7.Columns.Item(1).Caption

```

```

comboReferencia.DataSource = ds7
SQL.dataR.Close()

strSQL = " SELECT 1,' ' as codigo,' ' as nombre " & _
        " UNION " & _
        " SELECT 2,cast(ase_codigo as char) as
codigo,TRIM(CONCAT(ase_apellido,' ',ase_nombre)) as nombre " & _
        " FROM asesor " & _
        " ORDER BY 1,nombre "
SQL.Ejecutar(strSQL)
ds9.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboAsesor.DisplayMember = ds9.Columns.Item(2).Caption
comboAsesor.ValueMember = ds9.Columns.Item(1).Caption
comboAsesor.DataSource = ds9
SQL.dataR.Close()

htMarcas.Clear()
strSQL = " SELECT DISTINCT marca.mar_codigo as codigo,mod_codigo as
codigomod " & _
        " FROM marca " & _
        " INNER JOIN modelo " & _
        " ON marca.mar_codigo=modelo.mar_codigo " & _
        " ORDER BY mar_nombre,mod_nombre "
SQL.Ejecutar(strSQL)
While SQL.dataR.Read
    htMarcas.Add(Valor(SQL.dataR, 1), Valor(SQL.dataR, 0))
End While
SQL.dataR.Close()

htAseguradoras.Clear()
strSQL = " SELECT DISTINCT aseguradora.asg_codigo as
codigo,con_codigo as codigomod " & _
        " FROM aseguradora " & _
        " INNER JOIN contacto_aseguradora " & _
        " ON aseguradora.asg_codigo=contacto_aseguradora.asg_codigo
" & _
        " ORDER BY asg_nombre,con_nombre "
SQL.Ejecutar(strSQL)
While SQL.dataR.Read
    htAseguradoras.Add(Valor(SQL.dataR, 1), Valor(SQL.dataR, 0))
End While
SQL.dataR.Close()
End Sub

Private Sub vsfgDetalle_AfterEdit(ByVal sender As Object, ByVal e As
C1.Win.C1FlexGrid.RowColEventArgs) Handles vsfgDetalle.AfterEdit
    If e.Col = 1 Then

        If Not cmb.SelectedValue Is Nothing Then
            vsfgDetalle.Item(e.Row, 9) = cmb.SelectedValue
            vsfgDetalle.Item(e.Row, 2) =
htGruposProductos(htGrupos(cmb.SelectedValue))
            vsfgDetalle.Item(e.Row, 10) = htGrupos(cmb.SelectedValue)
        Else
            vsfgDetalle.Item(e.Row, 9) = ""
            vsfgDetalle.Item(e.Row, 2) = ""
        End If
    End If
End Sub

```

```

        vsfgDetalle.Item(e.Row, 10) = ""
    End If

    For i As Integer = 1 To vsfgDetalle.Rows.Count - 1
        If i <> e.Row Then
            If vsfgDetalle.Item(i, 9) = vsfgDetalle.Item(e.Row, 9)
And vsfgDetalle.Item(i, 9) <> "" Then
                MsgBox("El producto ya ha sido ingresado",
MsgBoxStyle.Information, Me.Text)
                vsfgDetalle.Item(e.Row, 1) = ""
                vsfgDetalle.Item(e.Row, 9) = ""
                vsfgDetalle.Item(e.Row, 2) = ""
                vsfgDetalle.Item(e.Row, 10) = ""
            End If
        End If
    Next

    strSQL = " SELECT IF(pro_costo IS NULL OR
pro_costo='',pro_costo,pro_costo) as precio,pro_iva " & _
        " FROM producto " & _
        " WHERE pro_codigo='" & vsfgDetalle.Item(e.Row, 9) & "' "

    SQL.Ejecutar(strSQL)
    If SQL.dataR.HasRows Then
        SQL.dataR.Read()
        vsfgDetalle.Item(e.Row, 4) = Numero("1")
        vsfgDetalle.Item(e.Row, 5) = Numero("0.00")
        vsfgDetalle.Item(e.Row, 6) = Numero("0.00")
        vsfgDetalle.Item(e.Row, 3) = Valor(SQL.dataR, 0)
        vsfgDetalle.Item(e.Row, 8) = Valor(SQL.dataR, 1)
        vsfgDetalle.Item(e.Row, 7) =
FormatNumber((vsfgDetalle.Item(e.Row, 3) * vsfgDetalle.Item(e.Row, 4)) -
vsfgDetalle.Item(e.Row, 5), 2)
    Else
        vsfgDetalle.Item(e.Row, 4) = Numero("1")
        vsfgDetalle.Item(e.Row, 5) = Numero("0.00")
        vsfgDetalle.Item(e.Row, 6) = Numero("0.00")
        vsfgDetalle.Item(e.Row, 3) = Numero("0.00")
        vsfgDetalle.Item(e.Row, 8) = "0"
        vsfgDetalle.Item(e.Row, 7) = Numero("0.00")
    End If
    SQL.dataR.Close()
    ElseIf e.Col = 3 Or e.Col = 4 Or e.Col = 5 Or e.Col = 6 Then
        If vsfgDetalle.Item(e.Row, e.Col).ToString <> "" Then
            vsfgDetalle.Item(e.Row, e.Col) =
Numero(vsfgDetalle.Item(e.Row, e.Col))
        End If
        If e.Col = 3 Then
            vsfgDetalle.Item(e.Row, e.Col) =
FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 2)
            If CDb1(vsfgDetalle.Item(e.Row, 6)) > 0 And
CDbl(vsfgDetalle.Item(e.Row, 5)) = 0 Then
                vsfgDetalle.Item(e.Row, 5) =
FormatNumber(FormatNumber(FormatNumber(vsfgDetalle.Item(e.Row, 3), 2) *
FormatNumber(vsfgDetalle.Item(e.Row, 4), 2), 2) *
(FormatNumber(vsfgDetalle.Item(e.Row, 6), 4) / 100), 2)
            End If
        End If
    End If

```

```

        End If
    ElseIf e.Col = 5 Then
        If FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 2) = 0 Then
            vsfgDetalle.Item(e.Row, e.Col + 1) = FormatNumber("0",
2)

            Else
                vsfgDetalle.Item(e.Row, e.Col + 1) =
FormatNumber((FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 2) * 100) /
FormatNumber((FormatNumber(vsfgDetalle.Item(e.Row, 3), 2) *
FormatNumber(vsfgDetalle.Item(e.Row, 4), 2)), 2), 2)
            End If

        ElseIf e.Col = 6 Then
            If FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 2) = 0 Then
                vsfgDetalle.Item(e.Row, e.Col - 1) = FormatNumber("0",
2)

            Else
                vsfgDetalle.Item(e.Row, e.Col - 1) =
FormatNumber(FormatNumber(FormatNumber(vsfgDetalle.Item(e.Row, 3), 2) *
FormatNumber(vsfgDetalle.Item(e.Row, 4), 2), 2) *
(FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 2) / 100), 2)
            End If
        End If

        If e.Col = 4 Or e.Col = 6 Then
            vsfgDetalle.Item(e.Row, e.Col) =
FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 0)
        Else
            vsfgDetalle.Item(e.Row, e.Col) =
FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 0)
        End If
        vsfgDetalle.Item(e.Row, 7) =
FormatNumber((vsfgDetalle.Item(e.Row, 3) * vsfgDetalle.Item(e.Row, 4)) -
vsfgDetalle.Item(e.Row, 5), 2)
        ElseIf e.Col = 2 Then
            If Not cmbG.Selected Value Is Nothing Then
                vsfgDetalle.Item(e.Row, 10) = cmbG.Selected Value
            Else
                vsfgDetalle.Item(e.Row, 10) = ""
            End If
        End If
        If e.Row = vsfgDetalle.Rows.Count - 1 And vsfgDetalle.Item(e.Row,
1) <> "" And vsfgDetalle.Item(e.Row, 3) <> 0 And vsfgDetalle.Item(e.Row, 4)
<> 0 Then
            vsfgDetalle.AddItem("")
            vsfgDetalle.Item(vsfgDetalle.Rows.Count - 1, 0) =
vsfgDetalle.Rows.Count - 1
            vsfgDetalle.SetCellImage(vsfgDetalle.Rows.Count - 1, 0,
ListaImagenes.Images(0))
        End If
        calcularTotal()
    End Sub

    Private Sub calcularTotal()
        Dim subtotal As Double, dcto As Double, iva12 As Double, iva0 As
Double, total As Double

```

```

        For i As Integer = 1 To vsfgDetalle.Rows.Count - 1
            If vsfgDetalle.Item(i, 1) <> Nothing OrElse vsfgDetalle.Item(i,
1) <> "" Then
                If vsfgDetalle.Item(i, 8) = True Then

                    checkIva.Checked = True

                    subtotal = subtotal +
FormatNumber(CDbl(vsfgDetalle.Item(i, 7)) + CDbl(vsfgDetalle.Item(i, 5)),
2)

                Else
                    iva0 = iva0 + FormatNumber(CDbl(vsfgDetalle.Item(i, 7))
+ CDbl(vsfgDetalle.Item(i, 5)), 2)
                End If
                dcto = dcto + FormatNumber(vsfgDetalle.Item(i, 5), 2)
            End If
        Next

        If checkIva.Checked Then
            iva12 = subtotal * (CDbl(strIVA) / 100)
            total = subtotal + iva0 + iva12 - dcto
        Else
            iva12 = 0
            total = subtotal + iva0 - dcto
        End If
        txtSubtotal.Text = FormatNumber(subtotal, 2)
        txtIVA.Text = FormatNumber(iva12, 2)
        txtIVA0.Text = FormatNumber(iva0, 2)
        txtDcto.Text = FormatNumber(dcto, 2)
        txtTotal.Text = FormatNumber(total, 2)
    End Sub

    Private Sub vsfgDetalle_BeforeEdit(ByVal sender As Object, ByVal e As
C1.Win.C1FlexGrid.RowColEventArgs) Handles vsfgDetalle.BeforeEdit
        If comboCliente.Text = "" Then
            e.Cancel = True
        ElseIf e.Col = 7 Then
            e.Cancel = True
        ElseIf e.Col = 2 Then
            If vsfgDetalle.Item(e.Row, 1) <> "" And vsfgDetalle.Item(e.Row,
9) <> "" Then
                e.Cancel = True
            End If
        End If
    End Sub

    Private Sub vsfgDetalle_MouseDown(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventHandler) Handles vsfgDetalle.MouseDown
        If e.Button <> Windows.Forms.MouseButtons.Left Then Exit Sub

        Dim r As Integer, c As Integer
        r = vsfgDetalle.MouseRow
        c = vsfgDetalle.MouseCol
        If r < 0 Or c < 0 Then Exit Sub
        If (c <> 0 Or r = (vsfgDetalle.Rows.Count - 1)) Then Exit Sub

```

```

        If MsgBox("Desea eliminar la fila " & r.ToString & "?",
MsgBoxStyle.Question + MsgBoxStyle.YesNo, "Descripción de Tabajos") =
MsgBoxResult.Yes Then
            vsfgDetalle.RemoveItem(r)
        Else
            vsfgDetalle.SetCellImage(r, c, ListaImágenes.Imágenes(0))
        End If
        calcularTotal()
    End Sub

```

```

    Private Sub comboAuto_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
comboAuto.SelectedIndexChanged
        txtAuto.Text = comboAuto.SelectedValue

        If comboAuto.SelectedValue = "" Then
            txtAuto.Text = "0"
        End If

```

```

    If Cargado = True Then
        If comboAuto.SelectedIndex > 0 Then
            cargarDatosAuto()
            comboMarca.Enabled = False
            comboModelo.Enabled = False
            comboColor.Enabled = False
            txtAno.Enabled = False
            txtMotor.ReadOnly = True
            txtChasis.ReadOnly = True
            txtPlaca.ReadOnly = True
        Else
            comboMarca.SelectedIndex = 0
            comboModelo.SelectedIndex = 0
            comboColor.SelectedIndex = 0
            txtAno.Text = ""
            txtMotor.Text = ""
            txtChasis.Text = ""
            txtPlaca.Text = ""

            comboMarca.Enabled = True
            comboModelo.Enabled = True
            comboColor.Enabled = True
            txtAno.Enabled = True
            txtMotor.ReadOnly = False
            txtChasis.ReadOnly = False
            txtPlaca.ReadOnly = False
        End If
    End If
End Sub

```

```

    Private Sub comboCliente_DropDownClosed(ByVal sender As Object, ByVal e
As System.EventArgs) Handles comboCliente.DropDownClosed
        If Cargado = True Then

            txtContacto.Text = comboCliente.SelectedValue
            If txtContacto.Text = "" Then

```

```

        txtListaPrecio.Text = ""
        LimpiarNuevo(False)
    Else
        txtListaPrecio.Text = ht.Item(comboCliente.SelectedVale)
        LimpiarNuevo(True)
    End If
End If
End Sub

Private Sub comboCliente_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles comboCliente.LostFocus
    If Cargado = True Then
        txtContacto.Text = comboCliente.SelectedVale
        If txtContacto.Text = "" Then
            txtListaPrecio.Text = ""
            LimpiarNuevo(False)
        Else
            txtListaPrecio.Text = ht.Item(comboCliente.SelectedVale)
            LimpiarNuevo(True)
        End If
    End If
End Sub

Private Sub comboMarca_DropDownClosed(ByVal sender As Object, ByVal e
As System.EventArgs) Handles comboMarca.DropDownClosed
    comboModelo.SelectedIndex = 0
End Sub

Private Sub comboMarca_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles comboMarca.LostFocus
    Dim ds3 As New DataTable

    strSQL = " SELECT 1,'% ' as codigo,'Seleccione uno...' as nombre " &
-
        " UNION " & _
        " SELECT 2,cast(mod_codigo as char) as codigo,mod_nombre
as nombre " & _
        " FROM modelo " & _
        " WHERE mar_codigo LIKE '" & comboMarca.SelectedVale & "'
" & _
        " ORDER BY 1,nombre "
    SQL.Ejecutar(strSQL)
    ds3.Load(SQL.dataR, LoadOption.OverwriteChanges)
    comboModelo.DisplayMember = ds3.Columns.Item(2).Caption
    comboModelo.ValueMember = ds3.Columns.Item(1).Caption
    comboModelo.DataSource = ds3
    SQL.dataR.Close()

End Sub

Private Sub comboModelo_DropDownClosed(ByVal sender As Object, ByVal e
As System.EventArgs) Handles comboModelo.DropDownClosed
    If htMarcas.Contains(comboModelo.SelectedVale) Then
        comboMarca.SelectedVale =
htMarcas.Item(comboModelo.SelectedVale)
    End If
End Sub

```



```

End Sub

Private Sub Limpiar()
    ''''CODIGOOO
    Dim num As String = "1"
    strSQL = " SELECT COALESCE(MAX(ROUND(prf_numero))+1,1) AS num " & _
        " FROM proforma "
    SQL.Ejecutar(strSQL)

    If SQL.dataR.HasRows Then
        num = Valor(SQL.dataR, 0)
    End If
    SQL.dataR.Close()
    txtNo.Text = CInt(num)
    strSQL = " SELECT pro_codigo as codigo,pro_nombre as nombre " & _
        " FROM producto "
    SQL.Ejecutar(strSQL)
    htProductos.Clear()
    While SQL.dataR.Read
        htProductos.Add(SQL.dataR.GetValue(0), SQL.dataR.GetValue(1))
    End While
    SQL.dataR.Close()
    strSQL = " SELECT pro_codigo as codigo,gru_pro_codigo as nombre " & _
        " FROM producto "
    SQL.Ejecutar(strSQL)
    htGrupos.Clear()
    While SQL.dataR.Read
        htGrupos.Add(SQL.dataR.GetValue(0), SQL.dataR.GetValue(1))
    End While
    SQL.dataR.Close()
    strSQL = " SELECT gru_pro_codigo as codigo,gru_pro_nombre as nombre " & _
        " FROM grupo_producto "
    SQL.Ejecutar(strSQL)
    htGruposProductos.Clear()
    While SQL.dataR.Read
        htGruposProductos.Add(SQL.dataR.GetValue(0),
SQL.dataR.GetValue(1))
    End While
    SQL.dataR.Close()

    cargarProductos()

    comboAseguradora.SelectedValue = ""
    comboReferencia.SelectedValue = ""
    comboReferencia.Text = ""
    txtPoliza.Text = ""
    txtAseguradora.Text = "0"
    txtAuto.Text = "0"
    comboAsegurado.SelectedValue = ""
    comboCliente.SelectedValue = ""
    txtObservaciones.Text = ""
    vsfgTipo.Cols(0).Item(0) = 1
    vsfgTipo.Item(0, 1) = ""
    vsfgTipo.Focus()

```

```

        LimpiarNuevo(False)
    End Sub

Private Sub LimpiarNuevo(ByVal tipo As Boolean)
    comboCliente.Tag = tipo
    comboAuto.Enabled = tipo

    If tipo = True Then
        strSQL = " SELECT per_ruc,per_tipo,ase_nombre " & _
                " FROM persona " & _
                " WHERE per_codigo='" & txtContacto.Text & "' " & _
                " AND per_que='C'"
        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        txtRuc.Text = Valor(SQL.dataR, 0)
        comboTipo.SelectedValue = Valor(SQL.dataR, 1)
        comboAsesor.SelectedValue = Valor(SQL.dataR, 2)
        SQL.dataR.Close()
    Else
        txtRuc.Text = ""
        txtNombre.Text = ""
        comboTipo.SelectedValue = "N"
        comboAsesor.SelectedValue = 0
    End If
    vsfgDetalle.Rows.Count = 1
    vsfgDetalle.Rows.Count = 2
    calcularTotal()
    PonerBotones()
    txtRuc.ReadOnly = tipo
    comboTipo.Enabled = Not tipo
    txtNombre.Visible = Not tipo
    lblNombre2.Visible = Not tipo
    cargarAuto()
End Sub

Private Sub cargarAuto()
    Dim ds0 As New DataTable
    strSQL = " SELECT 1,' ' as codigo,'Nuevo Auto...' as nombre " & _
            " UNION " & _
            " SELECT 2,cast(aut_codigo as char) as codigo,"
    TRIM(CONCAT(mar_nombre,' ',mod_nombre,' - ',cast(aut_ano as char),'
    (' ,aut_placa,'))) as nombre " & _
            " FROM auto " & _
            " INNER JOIN modelo " & _
            " ON modelo.mod_codigo=auto.mod_codigo " & _
            " INNER JOIN marca " & _
            " ON modelo.mar_codigo=marca.mar_codigo " & _
            " WHERE auto.per_codigo='" & txtContacto.Text & "' " & _
            " ORDER BY 1,nombre "
    SQL.Ejecutar(strSQL)
    ds0.Load(SQL.dataR, LoadOption.OverwriteChanges)
    comboAuto.DisplayMember = ds0.Columns.Item(2).Caption
    comboAuto.ValueMember = ds0.Columns.Item(1).Caption
    comboAuto.DataSource = ds0
    If comboAuto.SelectedValue = "" Then

```

```

        txtAuto.Text = "0"
    End If

    SQL.dataR.Close()
End Sub

Private Sub cargarDatosAuto()
    strSQL = " SELECT
auto.mod_codigo,mar_codigo,col_codigo,aut_ano,aut_placa,aut_motor,aut_chasis " & _
        " FROM auto " & _
        " INNER JOIN modelo " & _
        " ON modelo.mod_codigo=auto.mod_codigo " & _
        " WHERE per_codigo='" & txtContacto.Text & "'" & _
        " AND aut_codigo='" & txtAuto.Text & "'"
    SQL.Ejecutar(strSQL)
    SQL.dataR.Read()
    comboMarca.SelectedValue = Valor(SQL.dataR, 1)
    comboModelo.SelectedValue = Valor(SQL.dataR, 0)
    comboColor.SelectedValue = Valor(SQL.dataR, 2)
    txtAno.Text = Valor(SQL.dataR, 3)
    txtMotor.Text = Valor(SQL.dataR, 5)
    txtChasis.Text = Valor(SQL.dataR, 6)
    txtPlaca.Text = Valor(SQL.dataR, 4)
    SQL.dataR.Close()
End Sub

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If comprobarDatos() = True Then
        Dim codigoCliente As String = comboCliente.SelectedValue
        Dim codigoAuto As String = comboAuto.SelectedValue
        Dim codigoOrden As String

        If CBool(comboCliente.Tag) = False Then
            strSQL = " SELECT
cast(CONCAT('C',LPAD(COALESCE(MAX(RIGHT(per_codigo,LENGTH(per_codigo)-
1))+1,1),4,'0')) as char) as codigo " & _
                " FROM persona " & _
                " WHERE per_que='C' "
            SQL.Ejecutar(strSQL)
            If SQL.dataR.HasRows Then
                SQL.dataR.Read()
                codigoCliente = SQL.dataR.GetValue(0).ToString
                SQL.dataR.Close()
            End If
            strSQL = " INSERT INTO
persona(per_que,per_codigo,per_tipo,per_ruc," & _
"per_apellido,per_nombre,per_direccion,per_telefono,per_celular," & _
"per_ciudad,ase_nombre) VALUES('C','" & _
codigoCliente & "','" & comboTipo.SelectedValue &
"','" & _
Trim(txtRuc.Text) & "','" & _
Texto(comboCliente.Text) & "','" & _
Texto(txtNombre.Text) & "','" & _

```

```

        "" & "','" & _
        "" & "','" & _
        "" & "','0,'" & comboAsesor.DisplayMember & "'"
SQL.Ejecutar(strSQL, "L", "I")
End If

If comboAuto.DisplayMember = "Nuevo Auto..." Then
    If txtAno.Text = "" Then
        txtAno.Text = "0"
    End If
    strSQL = " INSERT INTO auto(per_codigo,mod_codigo," & _
        "aut_ano,aut_placa,col_codigo,aut_chasis,aut_motor)
VALUES('" & _
        codigoCliente & "','" & _
        comboModelo.SelectedValue & "','" & _
        Texto(txtAno.Text) & "','" & _
        Texto(txtPlaca.Text) & "','" & _
        Texto(txtColor.Text) & "','" & _
        Texto(txtChasis.Text) & "','" & _
        Texto(txtMotor.Text) & "'"
SQL.Ejecutar(strSQL, "L", "I")
strSQL = " SELECT max(aut_codigo) " & _
        " FROM auto " & _
        " WHERE per_codigo='" & codigoCliente & "'"
SQL.Ejecutar(strSQL)
If SQL.dataR.HasRows Then
    SQL.dataR.Read()
    codigoAuto = Valor(SQL.dataR, 0)
End If
SQL.dataR.Close()
End If

If vsfgTipo.Item(0, 0).ToString = "2" Then
    codigoOrden = vsfgTipo.Item(0, 1).ToString
    strSQL = " DELETE FROM proforma " & _
        " WHERE prf_codigo='" & codigoOrden & "'"
SQL.Ejecutar(strSQL, "L", "E")
strSQL = " DELETE FROM detalle_proforma " & _
        " WHERE prf_codigo='" & codigoOrden & "'"
SQL.Ejecutar(strSQL, "L", "E")
Else
    Dim Num As String = "1"
    strSQL = " SELECT
cast(COALESCE(RIGHT(MAX(prf_codigo+1),7),1) as char) AS num " & _
        " FROM proforma "
SQL.Ejecutar(strSQL)

    If SQL.dataR.HasRows Then
        Num = Valor(SQL.dataR, 0)
    End If
    SQL.dataR.Close()
    codigoOrden = Num.PadLeft(7, "0")
End If
'Ingresa la Proforma
'**** CABECERA

```

```

strSQL = " INSERT INTO proforma(prf_codigo,prf_numero," & _
"prf_fecha,est_prf_codigo,per_codigo,prf_destinatario,asg_codigo," & _
"prf_poliza,prf_asegurado,aut_codigo,ase_codigo,prf_perdcto,prf_subtotal,pr
f_subtotal_o,prf_dcto," & _
"prf_impuesto,prf_total,prf_observaciones) VALUES(' " &
_
codigoOrden & "',' " & _
Trim(txtNo.Text) & "',' " & _
Format(dtpFecha.Value, "yyyy-MM-dd") & "','1,' " & _
codigoCliente & "',' " & _
Texto(comboReferencia.Text) & "',' " & _
Texto(txtAseguradora.Text) & "',' " & _
Texto(txtPoliza.Text) & "',' " & _
Texto(comboAsegurado.Text) & "',' " & _
codigoAuto & "',' " & _
comboAsesor.SelectedValue & "',' " & _
NBDD(txtPerDcto.Text) & "',' " & _
NBDD(txtSubtotal.Text) & "',' " & _
NBDD(txtIVA0.Text) & "',' " & _
NBDD(txtDcto.Text) & "',' " & _
NBDD(txtIVA.Text) & "',' " & _
NBDD(txtTotal.Text) & "',' " & _
Texto(txtObservaciones.Text) & "')"
SQL.Ejecutar(strSQL, "L", "I")

'*****DETALLE
Dim codProducto As String, codGrupo As String
For i As Integer = 1 To vsfgDetalle.Rows.Count - 1
    If vsfgDetalle.Item(i, 1) <> "" Then
        codProducto = vsfgDetalle.Item(i, 9)
        If codProducto = "" Then
            strSQL = " SELECT
COALESCE(MAX(ROUND(pro_codigo))+1,1) FROM producto " & _
" WHERE pro_tipo='S' "
            SQL.Ejecutar(strSQL)
            SQL.dataR.Read()
            codProducto = CInt(Valor(SQL.dataR, 0))
            SQL.dataR.Close()
            codGrupo = vsfgDetalle.Item(i, 10)
            If codGrupo = "" Then
                strSQL = " SELECT gru_pro_codigo " & _
" FROM grupo_producto " & _
" ORDER BY gru_pro_codigo LIMIT 1 "
                SQL.Ejecutar(strSQL)
                SQL.dataR.Read()
                codGrupo = Valor(SQL.dataR, 0)
                SQL.dataR.Close()
            End If

            strSQL = " INSERT INTO
producto(pro_tipo,pro_codigo,uni_codigo," & _
"
pro_nombre,pro_marca,gru_pro_codigo,pro_costo,pro_iva,pro_descripcion)
VALUES(' " & _

```

```

                                "'S','" & codProducto & "','U','" & _
                                Texto(vsfgDetalle.Item(i, 1)) & "','','" &
codGrupo & "',''" & _
                                NBDD(vsfgDetalle.Item(i, 3)) & "',''" & _
                                Math.Abs(CInt(vsfgDetalle.Item(i, 8))) &
"'','')"
                                SQL.Ejecutar(strSQL, "L", "I")
                                strSQL = " INSERT INTO
existencia(pro_codigo,exi_cantidad) VALUES('" & _
                                codProducto & "','0) "
                                SQL.Ejecutar(strSQL, "L", "I")
                                End If
                                strSQL = " INSERT INTO detalle_proforma(prf_codigo," &
_
                                "pro_codigo,det_prf_cantidad,det_prf_precio,det_prf_dcto," & _
                                "det_prf_dctop,det_prf_total,det_prf_iva) VALUES('"
& _
                                codigoOrden & "',''" & _
                                codProducto & "',''" & _
                                NBDD(vsfgDetalle.Item(i, 4)) & "',''" & _
                                NBDD(vsfgDetalle.Item(i, 3)) & "',''" & _
                                NBDD(vsfgDetalle.Item(i, 5)) & "',''" & _
                                NBDD(vsfgDetalle.Item(i, 6)) & "',''" & _
                                NBDD(vsfgDetalle.Item(i, 7)) & "',''" & _
                                Math.Abs(CInt(vsfgDetalle.Item(i, 8))) & "'')"
                                SQL.Ejecutar(strSQL, "L", "I")
                                End If
                                Next
                                MsgBox("Se ha ingresado la Proforma No. " & Trim(txtNo.Text) &
" correctamente", MsgBoxStyle.Information, Me.Text)
                                Me.Cursor = Cursors.WaitCursor
                                Me.Cursor = Cursors.Default
                                Limpiar()
                                End If
                                End Sub

                                Private Function comprobarDatos() As Boolean
                                If Trim(txtNo.Text) = "" Then
                                MsgBox("Ingrese el número de proforma",
MsgBoxStyle.Information, Me.Text)
                                txtNo.Focus()
                                Return False
                                ElseIf comboCliente.Text = "" Then
                                MsgBox("Seleccione un cliente", MsgBoxStyle.Information,
Me.Text)
                                comboAsesor.Focus()
                                Return False
                                ElseIf comboAsesor.SelectedIndex <= 0 Then
                                MsgBox("Seleccione un asesor", MsgBoxStyle.Information,
Me.Text)
                                comboAsesor.Focus()
                                Return False
                                ElseIf comboMarca.SelectedIndex <= 0 Then
                                MsgBox("Seleccione una marca del auto",
MsgBoxStyle.Information, Me.Text)

```

```

        comboMarca.Focus()
        Return False
    ElseIf comboModelo.SelectedIndex <= 0 Then
        MsgBox("Seleccione un modelo del auto",
MsgBoxStyle.Information, Me.Text)
        comboModelo.Focus()
        Return False
    ElseIf vsfgTipo.Item(0, 0).ToString = "1" Then
        Return True
    Else
        Return True
    End If
End Function

Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub

Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
    limpiar()
End Sub

Private Sub PonerBotones()
    Dim i As Integer
    Try
        For i = 1 To (vsfgDetalle.Rows.Count - 1)
            vsfgDetalle.Item(i, 0) = i
            vsfgDetalle.SetCellImage(i, 0, ListaImagenes.Images(0))
        Next i
    Catch ex As Exception
    End Try
End Sub

Private Sub vsfgTipo_AfterEdit(ByVal sender As Object, ByVal e As
Cl.Win.ClFlexGrid.RowColEventArgs) Handles vsfgTipo.AfterEdit
    If e.Col = 0 Then
        If vsfgTipo.Item(0, 0).ToString = "2" Then
            'Modificar Proforma
            strSQL = " SELECT prf_codigo as
codigo,TRIM(CONCAT(prf_numero,' - ',per_apellido,' ',per_nombre)) as nombre
" & _
                " FROM proforma " & _
                " INNER JOIN persona " & _
                " ON persona.per_codigo=proforma.per_codigo " & _
                " AND persona.per_que='C' " & _
                " WHERE est_prf_codigo=1 " & _
                " ORDER BY prf_numero "
            SQL.Ejecutar(strSQL)
            Dim extra As New Hashtable
            While SQL.dataR.Read
                extra.Add(SQL.dataR.GetValue(0), SQL.dataR.GetValue(1))
            End While
            SQL.dataR.Close()
            vsfgTipo.Cols(1).DataMap = extra
        End If
    End If
End Sub

```

```

Else
    Limpiar()
End If
ElseIf e.Col = 1 Then

    txtProforma.Text = vsfgTipo.Item(0, 1).ToString

    If vsfgTipo.Item(0, 0).ToString = "2" And vsfgTipo.Item(0,
1).ToString <> "" Then
        'Modificar Orden
        '*** CABECERA
        strSQL = " SELECT
prf_codigo,prf_fecha,prf_numero,per_codigo, " & _
                " prf_destinatario,asg_codigo,aut_codigo, " & _
                " prf_poliza,prf_asegurado,prf_subtotal, " & _
                "
prf_subtotal_o,prf_dcto,prf_impuesto,prf_total,prf_observaciones,ase_codigo
,prf_perdcto " & _
                " FROM proforma " & _
                " WHERE prf_codigo='" & txtProforma.Text & "' "
        SQLaux.Ejecutar(strSQL)
        While SQLaux.dataR.Read

            txtContacto.Text = Valor(SQLaux.dataR, 3)

           .dtpFecha.Value = Valor(SQLaux.dataR, 1)
            txtNo.Text = Valor(SQLaux.dataR, 2)
            comboCliente.SelectedValue = Valor(SQLaux.dataR, 3)
            LimpiarNuevo(True)
            comboReferencia.Text = Valor(SQLaux.dataR, 4)
            comboAseguradora.SelectedValue = Valor(SQLaux.dataR, 5)
            txtAuto.Text = Valor(SQLaux.dataR, 6)
            txtAsesor.Text = Valor(SQLaux.dataR, 15)
            comboAuto.SelectedValue = Valor(SQLaux.dataR, 6)
            txtPoliza.Text = Valor(SQLaux.dataR, 7)
            comboAsegurado.Text = Valor(SQLaux.dataR, 8)
            txtSubtotal.Text = Valor(SQLaux.dataR, 9)
            txtIVA0.Text = Valor(SQLaux.dataR, 10)
            txtDcto.Text = Valor(SQLaux.dataR, 11)
            txtIVA.Text = Valor(SQLaux.dataR, 12)
            txtTotal.Text = Valor(SQLaux.dataR, 13)
            txtObservaciones.Text = Valor(SQLaux.dataR, 14)
            comboAsesor.SelectedValue = Valor(SQLaux.dataR, 15)
        End While
        SQLaux.dataR.Close()

        Dim row As Integer = 1
        '*** DETALLE
        strSQL = " SELECT
pro_codigo,det_prf_precio,det_prf_cantidad, " & _
                "
det_prf_dcto,det_prf_dctop,det_prf_total,det_prf_iva,pro_codigo " & _
                " FROM detalle_proforma " & _
                " WHERE prf_codigo='" & vsfgTipo.Item(0,
1).ToString & "' "

```



```

        SQLaux.Ejecutar(strSQL)

        vsfgDetalle.Rows.Count = 1
        While SQLaux.dataR.Read()
            vsfgDetalle.Rows.Add()
            vsfgDetalle.Item(row, 0) = row
            vsfgDetalle.Item(row, 1) =
htProductos(Valor(SQLaux.dataR, 0))
            vsfgDetalle.Item(row, 2) =
htGruposProductos(htGrupos(Valor(SQLaux.dataR, 0)))
            vsfgDetalle.Item(row, 3) = Valor(SQLaux.dataR, 1)
            vsfgDetalle.Item(row, 4) = Valor(SQLaux.dataR, 2)
            vsfgDetalle.Item(row, 5) = Valor(SQLaux.dataR, 3)
            vsfgDetalle.Item(row, 6) = Valor(SQLaux.dataR, 4)
            vsfgDetalle.Item(row, 7) = Valor(SQLaux.dataR, 5)
            vsfgDetalle.Item(row, 8) = Valor(SQLaux.dataR, 6)
            vsfgDetalle.Item(row, 9) = Valor(SQLaux.dataR, 7)
            vsfgDetalle.Item(row, 10) =
htGrupos(Valor(SQLaux.dataR, 0))

            row = row + 1
        End While
        SQLaux.dataR.Close()

        vsfgDetalle.AddItem("")
        vsfgDetalle.Item(vsfgDetalle.Rows.Count - 1, 0) =
vsfgDetalle.Rows.Count - 1
        vsfgDetalle.SetCellImage(vsfgDetalle.Rows.Count - 1, 0,
ListaImagenes.Images(0))
        ElseIf vsfgTipo.Item(0, 1).ToString = "" Then
            Exit Sub
        End If
    End If
    PonerBotones()
    calcularTotal()
End Sub

Private Sub vsfgTipo_BeforeEdit(ByVal sender As System.Object, ByVal e
As Cl.Win.ClFlexGrid.RowColEventArgs) Handles vsfgTipo.BeforeEdit
    If e.Col = 1 Then
        If vsfgTipo.Item(0, 0).ToString = "1" Then
            e.Cancel = True
        End If
    End If
End Sub

Private Sub comboAseguradora_DropDownClosed(ByVal sender As Object,
ByVal e As System.EventArgs) Handles comboAseguradora.DropDownClosed
    Dim ds3 As New DataTable

    If comboAseguradora.DisplayMember = "" Then
        txtAseguradora.Text = "Hola"
    End If
    strSQL = " SELECT 1,' ' as codigo,' ' as nombre " & _

```

```

        " UNION " & _
        " SELECT 2, cast(con_codigo as char) as
codigo,TRIM(CONCAT(con_nombre,' ',con_apellido)) as nombre " & _
        " FROM contacto_aseguradora " & _
        " WHERE asg_codigo='" & comboAseguradora.SelectedValue &
"' " & _
        " ORDER BY 1,nombre "
SQL.Ejecutar(strSQL)
ds3.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboReferencia.DisplayMember = ds3.Columns.Item(2).Caption
comboReferencia.ValueMember = ds3.Columns.Item(1).Caption
comboReferencia.DataSource = ds3
SQL.dataR.Close()
comboReferencia.SelectedValue = 0
End Sub

Private Sub comboReferencia_DropDownClosed(ByVal sender As Object,
ByVal e As System.EventArgs) Handles comboReferencia.DropDownClosed
    If comboReferencia.SelectedValue Is Nothing Then

    Else
        If htAseguradoras.Contains(comboReferencia.SelectedValue) Then
            comboAseguradora.SelectedValue =
htAseguradoras.Item(comboReferencia.SelectedValue)
        Else
            comboAseguradora.SelectedIndex = 0
        End If
    End If
End Sub

Private Sub comboTipo_DropDownClosed(ByVal sender As Object, ByVal e As
System.EventArgs) Handles comboTipo.DropDownClosed
    If comboTipo.SelectedValue = "N" Then
        txtNombre.Enabled = True
    Else
        txtNombre.Enabled = False
    End If
End Sub

Private Sub txtNo_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtNo.LostFocus

    Dim texto() As Char, pos As Integer, cadena As String, cadena2 As
String, cadena3 As Integer
    Dim Limpiado As String = ""
    txtNo.Text = txtNo.Text.ToUpper()
    texto = Aux.ToCharArray
    pos = txtNo.Text.ToUpper().LastIndexOfAny(texto)
    cadena2 = txtNo.Text.Substring(pos + 1)
    cadena3 = Cint(txtNo.Text.Substring(pos + 1) + 1)
    cadena = txtNo.Text.Substring(0, pos + 1) &
txtNo.Text.Substring(pos + 1, Len(cadena2) - Len(cadena3.ToString)) &
cadena3
    Dim cuantos As Integer = 0
    strSQL = " SELECT COUNT(*) " & _
        " FROM proforma " & _

```

```

        " WHERE UCASE(prf_numero)='" & UCase(txtNo.Text) & "' "
SQL.Ejecutar(strSQL)
If SQL.dataR.HasRows Then
    SQL.dataR.Read()
    cuantos = CInt(SQL.dataR.GetValue(0))
    SQL.dataR.Close()
End If

    If cuantos > 0 And InStr(vsfgTipo.GetDataDisplay(0, 1), txtNo.Text,
CompareMethod.Text) = 0 Then
        ErrorProvider.SetError(txtNo, "El No. de Proforma ya existe, se
sugiere utilizar: " & cadena)
    Else
        ErrorProvider.SetError(txtNo, "")
    End If
End Sub

Private Sub txtRuc_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtRuc.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub txtAno_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtAno.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub txtNombre_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtNombre.LostFocus
    vsfgDetalle.Rows.Count = 1
    vsfgDetalle.Rows.Count = 2
    calcularTotal()
    PonerBotones()
End Sub

Private Sub comboModelo_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
comboModelo.SelectedIndexChanged
    txtModelo.Text = comboModelo.SelectedValue
End Sub

Private Sub comboAsesor_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
comboAsesor.SelectedIndexChanged
    txtAsesor.Text = comboAsesor.SelectedValue
End Sub

Private Sub comboAseguradora_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
comboAseguradora.SelectedIndexChanged
    If comboAseguradora.SelectedValue = "" Then
        txtAseguradora.Text = "0"
    Else

```

```

        txtAseguradora.Text = comboAseguradora.SelectedValue
    End If
End Sub

Private Sub comboColor_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
comboColor.SelectedIndexChanged
    If comboColor.SelectedValue = "" Then
        txtColor.Text = "0"
    Else
        txtColor.Text = comboColor.SelectedValue
    End If
End Sub

End Class

```

Orden de Trabajo

```

Public Class frmOrdenTrabajo
    Private strSQL As String
    Private SQL As New clsSQL
    Private SQLaux As New clsSQL
    Private ht As New Hashtable
    Private htCategoria As New Hashtable
    Private htMarcas As New Hashtable
    Private Cargado As Boolean
    Private htProductos As New Hashtable
    Private htGruposProductos As New Hashtable
    Private htGrupos As New Hashtable
    Private cmb As New ComboBox
    Private cmbG As New ComboBox
    Private Aux As String = "ABCDEFGHIJKLMNÑOPQRSTUVWXYZ"
    Private Sub frmOrdenTrabajo_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        SQLaux.Inicializar(ConnMaster)
        Cargado = False

        strSQL = " SELECT tip_ord_codigo as codigo,tip_ord_nombre as nombre
" & _
                " FROM tipo_orden " & _
                " ORDER BY tip_ord_codigo "
        SQL.Ejecutar(strSQL)
        Dim extra As New Specialized.ListDictionary

        While SQL.dataR.Read
            extra.Add(SQL.dataR.GetValue(0), SQL.dataR.GetValue(1))
        End While
        SQL.dataR.Close()
        vsfgTipo.Cols(0).DataMap = extra
        vsfgTipo.Cols(0).Item(0) = 1

        cargarCombo()
        Limpiar()
        strSQL = " SELECT par_valor as valor " & _

```

```

        " FROM parametro " & _
        " WHERE par_codigo='IVA' "
SQL.Ejecutar(strSQL)
SQL.dataR.Read()
strIVA = Valor(SQL.dataR, 0)
SQL.dataR.Close()

lblIva.Text = "IVA " & Format(CDbl(strIVA), "#0.##") & "%"
Cargado = True
End Sub
Private Sub cargarProductos()
strSQL = " SELECT pro_codigo as codigo,pro_nombre as nombre " & _
        " FROM producto " & _
        " ORDER BY pro_nombre ASC "
SQL.Ejecutar(strSQL)

Dim ds0 As New DataTable

cmb.AutoCompleteMode = System.Windows.Forms.AutoCompleteMode.Append
cmb.AutoCompleteSource = AutoCompleteSource.ListItems
cmb.DropDownStyle = ComboBoxStyle.DropDown
cmb.FormattingEnabled = True
cmb.Name = "cmb"

ds0.Load(SQL.dataR, LoadOption.OverwriteChanges)
cmb.DisplayMember = ds0.Columns.Item(1).Caption
cmb.ValueMember = ds0.Columns.Item(0).Caption
cmb.DataSource = ds0
SQL.dataR.Close()
vsfgDetalle.Cols(1).Editor = cmb
strSQL = " SELECT gru_pro_codigo as codigo,gru_pro_nombre as nombre
" & _
        " FROM grupo_producto " & _
        " ORDER BY gru_pro_nombre ASC "
SQL.Ejecutar(strSQL)

Dim dsx As New DataTable

cmbG.AutoCompleteSource = AutoCompleteSource.ListItems
cmbG.DropDownStyle = ComboBoxStyle.DropDownList
cmbG.Name = "cmbG"

dsx.Load(SQL.dataR, LoadOption.OverwriteChanges)
cmbG.DisplayMember = dsx.Columns.Item(1).Caption
cmbG.ValueMember = dsx.Columns.Item(0).Caption
cmbG.DataSource = dsx
SQL.dataR.Close()
vsfgDetalle.Cols(2).Editor = cmbG

End Sub

Private Sub cargarCombo()
Dim ds0 As New DataTable, ds1 As New DataTable
Dim ds2 As New DataTable, ds3 As New DataTable
Dim ds4 As New DataTable, ds5 As New DataTable, ds6 As New
DataTable

```

```

strSQL = " SELECT 'N' as codigo,'Natural' as nombre " & _
        " UNION " & _
        " SELECT 'J' as codigo,'Jurídica' as nombre "
SQL.Ejecutar(strSQL)
ds6.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboTipo.DisplayMember = ds6.Columns.Item(1).Caption
comboTipo.ValueMember = ds6.Columns.Item(0).Caption
comboTipo.DataSource = ds6
SQL.dataR.Close()
strSQL = " SELECT 1,'' as codigo,'' as nombre,'' as per_apellido,''
as per_nombre " & _
        " UNION " & _
        " SELECT 2,cast(per_codigo as char) as
codigo,TRIM(CONCAT(per_apellido,' ',per_nombre)) as
nombre,per_apellido,per_nombre " & _
        " FROM persona " & _
        " WHERE per_que='C' " & _
        " ORDER BY 1,per_apellido,per_nombre "
SQL.Ejecutar(strSQL)
ds0.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboCliente.DisplayMember = ds0.Columns.Item(2).Caption
comboCliente.ValueMember = ds0.Columns.Item(1).Caption
comboCliente.DataSource = ds0
SQL.dataR.Close()

strSQL = " SELECT 1,'' as codigo,'' as nombre " & _
        " UNION " & _
        " SELECT 2,cast(ase_codigo as char) as
codigo,TRIM(CONCAT(ase_apellido,' ',ase_nombre)) as nombre " & _
        " FROM asesor " & _
        " ORDER BY 1,nombre "
SQL.Ejecutar(strSQL)
ds1.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboAsesor.DisplayMember = ds1.Columns.Item(2).Caption
comboAsesor.ValueMember = ds1.Columns.Item(1).Caption
comboAsesor.DataSource = ds1
SQL.dataR.Close()

strSQL = " SELECT 1,'% ' as codigo,'Seleccione una...' as nombre " &
-
        " UNION " & _
        " SELECT 2, cast(mar_codigo as char) as codigo,mar_nombre
as nombre " & _
        " FROM marca " & _
        " ORDER BY 1,nombre "
SQL.Ejecutar(strSQL)
ds2.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboMarca.DisplayMember = ds2.Columns.Item(2).Caption
comboMarca.ValueMember = ds2.Columns.Item(1).Caption
comboMarca.DataSource = ds2
SQL.dataR.Close()

strSQL = " SELECT 1,'% ' as codigo,'Seleccione uno...' as nombre " &
-
        " UNION " & _

```

```

        " SELECT 2,cast(mod_codigo as char) as codigo,mod_nombre
as nombre " & _
        " FROM modelo " & _
        " ORDER BY 1,nombre "
SQL.Ejecutar(strSQL)
ds3.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboModelo.DisplayMember = ds3.Columns.Item(2).Caption
comboModelo.ValueMember = ds3.Columns.Item(1).Caption
comboModelo.DataSource = ds3
SQL.dataR.Close()

strSQL = " SELECT 1,'' as codigo,'' as nombre " & _
        " UNION " & _
        " SELECT 2,cast(col_codigo as char) as codigo,col_nombre
as nombre " & _
        " FROM color " & _
        " ORDER BY 1,nombre "
SQL.Ejecutar(strSQL)
ds4.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboColor.DisplayMember = ds4.Columns.Item(2).Caption
comboColor.ValueMember = ds4.Columns.Item(1).Caption
comboColor.DataSource = ds4
SQL.dataR.Close()

htMarcas.Clear()
strSQL = " SELECT DISTINCT marca.mar_codigo as codigo,mod_codigo as
codigomod " & _
        " FROM marca " & _
        " INNER JOIN modelo " & _
        " ON marca.mar_codigo=modelo.mar_codigo " & _
        " ORDER BY mar_nombre,mod_nombre "
SQL.Ejecutar(strSQL)
While SQL.dataR.Read
    htMarcas.Add(Valor(SQL.dataR, 1), Valor(SQL.dataR, 0))
End While
SQL.dataR.Close()
End Sub

Private Sub vsfgDetalle_AfterEdit(ByVal sender As Object, ByVal e As
C1.Win.C1FlexGrid.RowColEventArgs) Handles vsfgDetalle.AfterEdit
    If e.Col = 1 Then
        If Not cmb.Selectedvalue Is Nothing Then
            vsfgDetalle.Item(e.Row, 9) = cmb.Selectedvalue
            vsfgDetalle.Item(e.Row, 2) =
htGruposProductos(htGrupos(cmb.Selectedvalue))
            vsfgDetalle.Item(e.Row, 10) = htGrupos(cmb.Selectedvalue)
        Else
            vsfgDetalle.Item(e.Row, 9) = ""
            vsfgDetalle.Item(e.Row, 2) = ""
            vsfgDetalle.Item(e.Row, 10) = ""
        End If
        For i As Integer = 1 To vsfgDetalle.Rows.Count - 1
            If i <> e.Row Then
                If vsfgDetalle.Item(i, 9) = vsfgDetalle.Item(e.Row, 9)
And vsfgDetalle.Item(i, 9) <> "" Then

```

```

        MsgBox("El producto ya ha sido ingresado",
MsgBoxStyle.Information, Me.Text)
        vsfgDetalle.Item(e.Row, 1) = ""
        vsfgDetalle.Item(e.Row, 9) = ""
        vsfgDetalle.Item(e.Row, 2) = ""
        vsfgDetalle.Item(e.Row, 10) = ""
    End If
End If
Next
strSQL = " SELECT IF(pro_costo IS NULL OR
pro_costo='',pro_costo,pro_costo) as precio,pro_iva " & _
        " FROM producto " & _
        " WHERE producto.pro_codigo='" &
vsfgDetalle.Item(e.Row, 9) & "' "

SQL.Ejecutar(strSQL)
If SQL.dataR.HasRows Then
    SQL.dataR.Read()
    vsfgDetalle.Item(e.Row, 4) = Numero("1")
    vsfgDetalle.Item(e.Row, 5) = Numero("0.00")
    vsfgDetalle.Item(e.Row, 6) = Numero("0.00")
    vsfgDetalle.Item(e.Row, 3) = Valor(SQL.dataR, 0)
    vsfgDetalle.Item(e.Row, 8) = Valor(SQL.dataR, 1)
    vsfgDetalle.Item(e.Row, 7) =
FormatNumber((vsfgDetalle.Item(e.Row, 3) * vsfgDetalle.Item(e.Row, 4)) -
vsfgDetalle.Item(e.Row, 5), 2)
    Else
        vsfgDetalle.Item(e.Row, 4) = Numero("1")
        vsfgDetalle.Item(e.Row, 5) = Numero("0.00")
        vsfgDetalle.Item(e.Row, 6) = Numero("0.00")
        vsfgDetalle.Item(e.Row, 3) = Numero("0.00")
        vsfgDetalle.Item(e.Row, 8) = "0"
        vsfgDetalle.Item(e.Row, 7) = Numero("0.00")
    End If
    SQL.dataR.Close()
    ElseIf e.Col = 3 Or e.Col = 4 Or e.Col = 5 Or e.Col = 6 Then
        If vsfgDetalle.Item(e.Row, e.Col).ToString <> "" Then
            vsfgDetalle.Item(e.Row, e.Col) =
Numero(vsfgDetalle.Item(e.Row, e.Col))
        End If
        If e.Col = 3 Then
            vsfgDetalle.Item(e.Row, e.Col) =
FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 2)
            If Cdbl(vsfgDetalle.Item(e.Row, 6)) > 0 And
Cdbl(vsfgDetalle.Item(e.Row, 5)) = 0 Then
                vsfgDetalle.Item(e.Row, 5) =
FormatNumber(FormatNumber(FormatNumber(vsfgDetalle.Item(e.Row, 3), 2) *
FormatNumber(vsfgDetalle.Item(e.Row, 4), 2), 2) *
(FormatNumber(vsfgDetalle.Item(e.Row, 6), 4) / 100), 2)
            End If
        ElseIf e.Col = 5 Then
            'Dcto. $
            If FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 2) = 0 Then
                vsfgDetalle.Item(e.Row, e.Col + 1) = FormatNumber("0",
2)
            Else

```



```

        vsfgDetalle.Item(e.Row, e.Col + 1) =
FormatNumber((FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 2) * 100) /
FormatNumber((FormatNumber(vsfgDetalle.Item(e.Row, 3), 2) *
FormatNumber(vsfgDetalle.Item(e.Row, 4), 2)), 2), 2)
        End If

    ElseIf e.Col = 6 Then
        'Dcto. %
        If FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 2) = 0 Then
            vsfgDetalle.Item(e.Row, e.Col - 1) = FormatNumber("0",
2)

        Else
            vsfgDetalle.Item(e.Row, e.Col - 1) =
FormatNumber(FormatNumber(FormatNumber(vsfgDetalle.Item(e.Row, 3), 2) *
FormatNumber(vsfgDetalle.Item(e.Row, 4), 2), 2) *
(FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 2) / 100), 2)
            End If
        End If

        If e.Col = 4 Or e.Col = 6 Then
            vsfgDetalle.Item(e.Row, e.Col) =
FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 0)
        Else
            vsfgDetalle.Item(e.Row, e.Col) =
FormatNumber(vsfgDetalle.Item(e.Row, e.Col), 0)
        End If
        vsfgDetalle.Item(e.Row, 7) =
FormatNumber((vsfgDetalle.Item(e.Row, 3) * vsfgDetalle.Item(e.Row, 4)) -
vsfgDetalle.Item(e.Row, 5), 2)
        ElseIf e.Col = 2 Then
            If Not cmbG.SelectedValue Is Nothing Then
                vsfgDetalle.Item(e.Row, 10) = cmbG.SelectedValue
            Else
                vsfgDetalle.Item(e.Row, 10) = ""
            End If
        End If
        If e.Row = vsfgDetalle.Rows.Count - 1 And vsfgDetalle.Item(e.Row,
1) <> "" And vsfgDetalle.Item(e.Row, 3) <> "" And vsfgDetalle.Item(e.Row,
4) <> "" Then
            vsfgDetalle.AddItem("")
            vsfgDetalle.Item(vsfgDetalle.Rows.Count - 1, 0) =
vsfgDetalle.Rows.Count - 1
            vsfgDetalle.SetCellImage(vsfgDetalle.Rows.Count - 1, 0,
ListaImagenes.Images(0))
        End If
        calcularTotal()
    End Sub

    Private Sub calcularTotal()
        Dim subtotal As Double, dcto As Double, iva12 As Double, iva0 As
Double, total As Double
        For i As Integer = 1 To vsfgDetalle.Rows.Count - 1
            If vsfgDetalle.Item(i, 1) <> Nothing OrElse vsfgDetalle.Item(i,
1) <> "" Then
                If vsfgDetalle.Item(i, 8) = True Then

```

```

        checkIva.Checked = True

        subtotal = subtotal +
FormatNumber(CDbl(vsfgDetalle.Item(i, 7)) + CDbl(vsfgDetalle.Item(i, 5)),
2)
            Else
                iva0 = iva0 + FormatNumber(CDbl(vsfgDetalle.Item(i, 7))
+ CDbl(vsfgDetalle.Item(i, 5)), 2)
            End If
            dcto = dcto + FormatNumber(vsfgDetalle.Item(i, 5), 2)
        End If
    Next

    If checkIva.Checked Then
        iva12 = subtotal * (CDbl(strIVA) / 100)
        total = subtotal + iva0 + iva12 - dcto
    Else
        iva12 = 0
        total = subtotal + iva0 - dcto
    End If

    txtSubtotal.Text = FormatNumber(subtotal, 2)
    txtIVA.Text = FormatNumber(iva12, 2)
    txtIVA0.Text = FormatNumber(iva0, 2)
    txtDcto.Text = FormatNumber(dcto, 2)
    txtTotal.Text = FormatNumber(total, 2)
End Sub

Private Sub vsfgDetalle_BeforeEdit(ByVal sender As Object, ByVal e As
Cl.Win.C1FlexGrid.RowColEventArgs) Handles vsfgDetalle.BeforeEdit
    If comboCliente.Text = "" Then
        e.Cancel = True
    ElseIf e.Col = 7 Then
        e.Cancel = True
    ElseIf e.Col = 2 Then
        If vsfgDetalle.Item(e.Row, 1) <> "" And vsfgDetalle.Item(e.Row,
9) <> "" Then
            e.Cancel = True
        End If
    End If
End Sub

Private Sub vsfgDetalle_MouseDown(ByVal sender As Object, ByVal e As
System.Windows.Forms.MouseEventHandler) Handles vsfgDetalle.MouseDown
    If e.Button <> Windows.Forms.MouseButtons.Left Then Exit Sub

    Dim r As Integer, c As Integer
    r = vsfgDetalle.MouseRow
    c = vsfgDetalle.MouseCol
    If r < 0 Or c < 0 Then Exit Sub
    If (c <> 0 Or r = (vsfgDetalle.Rows.Count - 1)) Then Exit Sub

    If MsgBox("Desea eliminar la fila " & r.ToString & "?",
MsgBoxStyle.Question + MsgBoxStyle.YesNo, "Descripción de Tabajos") =
MsgBoxResult.Yes Then
        vsfgDetalle.RemoveItem(r)

```

```

Else
    vsfgDetalle.SetCellImage(r, c, ListaImágenes.Imágenes(0))
End If
calcularTotal()
End Sub

Private Sub comboAuto_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
comboAuto.SelectedIndexChanged
    txtAuto.Text = comboAuto.SelectedValue

    If comboAuto.SelectedValue = "" Then
        txtAuto.Text = "0"
    End If

    If Cargado = True Then
        If comboAuto.SelectedIndex > 0 Then
            cargarDatosAuto()
            comboMarca.Enabled = False
            comboModelo.Enabled = False
            comboColor.Enabled = False
            txtAno.Enabled = False
            txtMotor.ReadOnly = True
            txtChasis.ReadOnly = True
            txtPlaca.ReadOnly = True
        Else
            comboMarca.SelectedIndex = 0
            comboModelo.SelectedIndex = 0
            comboColor.SelectedIndex = 0
            txtAno.Text = ""
            txtMotor.Text = ""
            txtChasis.Text = ""
            txtPlaca.Text = ""

            comboMarca.Enabled = True
            comboModelo.Enabled = True
            comboColor.Enabled = True
            txtAno.Enabled = True
            txtMotor.ReadOnly = False
            txtChasis.ReadOnly = False
            txtPlaca.ReadOnly = False
        End If
    End If
End Sub

Private Sub comboCliente_DropDownClosed(ByVal sender As Object, ByVal e
As System.EventArgs) Handles comboCliente.DropDownClosed
    If Cargado = True Then

        txtContacto.Text = comboCliente.SelectedValue
        If txtContacto.Text = "" Then
            txtListaPrecio.Text = ""
            LimpiarNuevo(False)
        Else
            txtListaPrecio.Text = ht.Item(comboCliente.SelectedValue)
        End If
    End If
End Sub

```

```

        LimpiarNuevo(True)
    End If
End If
End Sub

Private Sub comboCliente_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles comboCliente.LostFocus
    If Cargado = True Then

        txtContacto.Text = comboCliente.SelectedValue

        If txtContacto.Text = "" Then
            txtListaPrecio.Text = ""
            LimpiarNuevo(False)
        Else
            txtListaPrecio.Text = ht.Item(comboCliente.SelectedValue)
            LimpiarNuevo(True)
        End If
    End If
End Sub

Private Sub comboMarca_DropDownClosed(ByVal sender As Object, ByVal e
As System.EventArgs) Handles comboMarca.DropDownClosed
    comboModelo.SelectedIndex = 0
End Sub

Private Sub comboMarca_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles comboMarca.LostFocus
    Dim ds3 As New DataTable
    strSQL = " SELECT 1,'% ' as codigo,'Seleccione uno...' as nombre " &
-
        " UNION " & _
        " SELECT 2,cast(mod_codigo as char) as codigo,mod_nombre as
nombre " & _
        " FROM modelo " & _
        " WHERE mar_codigo LIKE '" & comboMarca.SelectedValue & "'
" & _
        " ORDER BY 1,nombre "
    SQL.Ejecutar(strSQL)
    ds3.Load(SQL.dataR, LoadOption.OverwriteChanges)
    comboModelo.DisplayMember = ds3.Columns.Item(2).Caption
    comboModelo.ValueMember = ds3.Columns.Item(1).Caption
    comboModelo.DataSource = ds3
    SQL.dataR.Close()
End Sub

Private Sub comboModelo_DropDownClosed(ByVal sender As Object, ByVal e
As System.EventArgs) Handles comboModelo.DropDownClosed
    If htMarcas.Contains(comboModelo.SelectedValue) Then
        comboMarca.SelectedValue =
htMarcas.Item(comboModelo.SelectedValue)
    End If
End Sub

Private Sub Limpiar()
    comboCliente.SelectedValue = ""

```

```

txtObservaciones.Text = ""

''''CODIGOOO
Dim num As String = "1"
strSQL = " SELECT COALESCE(MAX(ROUND(ord_tra_numero))+1,1) AS num "
& _
        " FROM orden_trabajo "

SQL.Ejecutar(strSQL)

If SQL.dataR.HasRows Then
    num = Valor(SQL.dataR, 0)
End If
SQL.dataR.Close()
txtNo.Text = CInt(num)
strSQL = " SELECT pro_codigo as codigo,pro_nombre as nombre " & _
        " FROM producto " & _
        " ORDER BY pro_nombre ASC "
SQL.Ejecutar(strSQL)
htProductos.Clear()
While SQL.dataR.Read
    htProductos.Add(SQL.dataR.GetValue(0), SQL.dataR.GetValue(1))
End While
SQL.dataR.Close()
strSQL = " SELECT pro_codigo as codigo,gru_pro_codigo as nombre " &
-
        " FROM producto "
SQL.Ejecutar(strSQL)
htGrupos.Clear()
While SQL.dataR.Read
    htGrupos.Add(SQL.dataR.GetValue(0), SQL.dataR.GetValue(1))
End While
SQL.dataR.Close()
strSQL = " SELECT gru_pro_codigo as codigo,gru_pro_nombre as nombre
" & _
        " FROM grupo_producto "
SQL.Ejecutar(strSQL)
htGruposProductos.Clear()
While SQL.dataR.Read
    htGruposProductos.Add(SQL.dataR.GetValue(0),
SQL.dataR.GetValue(1))
End While
SQL.dataR.Close()

cargarProductos()

vsfgTipo.Cols(0).Item(0) = 1
vsfgTipo.Item(0, 1) = ""
vsfgTipo.Focus()

lblSiniestro.Visible = False
txtSiniestro.Enabled = False
txtSiniestro.Visible = False

LimpiarNuevo(False)
End Sub

```

```

Private Sub LimpiarNuevo(ByVal tipo As Boolean, Optional ByVal todo As
Boolean = True)
    comboCliente.Tag = tipo
    comboAuto.Enabled = tipo
    txtSiniestro.Text = ""
    txtReferencia.Text = ""
    If tipo = True Then
        strSQL = " SELECT
per_telefono,per_celular,per_direccion,ase_nombre,per_ruc,per_tipo " & _
                " FROM persona " & _
                " WHERE per_codigo='" & comboCliente.SelectedValue & "'
" & _
                " AND per_que='C' "
        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        txtTelefono.Text = Valor(SQL.dataR, 0)
        txtCelular.Text = Valor(SQL.dataR, 1)
        txtDireccion.Text = Valor(SQL.dataR, 2)
        comboAsesor.SelectedValue = Valor(SQL.dataR, 3)
        txtRuc.Text = Valor(SQL.dataR, 4)
        comboTipo.SelectedValue = Valor(SQL.dataR, 5)
        SQL.dataR.Close()
    Else
        txtRuc.Text = ""
        txtTelefono.Text = ""
        txtCelular.Text = ""
        txtNombre.Text = ""
        txtDireccion.Text = ""
        comboAsesor.SelectedValue = 0
        comboTipo.SelectedValue = "N"
    End If
    vsfgDetalle.Rows.Count = 1
    vsfgDetalle.Rows.Count = 2
    calcularTotal()
    PonerBotones()

    txtTelefono.ReadOnly = tipo
    txtCelular.ReadOnly = tipo
    txtDireccion.ReadOnly = tipo
    txtRuc.ReadOnly = tipo

    comboTipo.Enabled = Not tipo
    txtNombre.Visible = Not tipo
    lblNombre2.Visible = Not tipo
    cargarAuto()

    comboAuto.Enabled = todo
    comboCliente.Enabled = todo

End Sub

Private Sub cargarAuto()
    Dim ds0 As New DataTable

```

```

        strSQL = " SELECT 1, '' as codigo, 'Nuevo Auto...' as nombre " & _
                " UNION " & _
                " SELECT 2, cast(aut_codigo as char) as
codigo, TRIM(CONCAT(mar_nombre, ' ', mod_nombre, ' - ', cast(aut_ano as char), '
(' , aut_placa, '))) as nombre " & _
                " FROM auto " & _
                " INNER JOIN modelo " & _
                " ON modelo.mod_codigo=auto.mod_codigo " & _
                " INNER JOIN marca " & _
                " ON modelo.mar_codigo=marca.mar_codigo " & _
                " WHERE auto.per_codigo='" & txtContacto.Text & "' " & _
                " ORDER BY 1, nombre "

SQL.Ejecutar(strSQL)
ds0.Load(SQL.dataR, LoadOption.OverwriteChanges)
comboAuto.DisplayMember = ds0.Columns.Item(2).Caption
comboAuto.ValueMember = ds0.Columns.Item(1).Caption
comboAuto.DataSource = ds0
SQL.dataR.Close()
End Sub

Private Sub cargarDatosAuto()
    strSQL = " SELECT
auto.mod_codigo, mar_codigo, col_codigo, aut_ano, aut_placa, aut_motor, aut_chasi
s " & _
            " FROM auto " & _
            " INNER JOIN modelo " & _
            " ON modelo.mod_codigo=auto.mod_codigo " & _
            " WHERE per_codigo='" & txtContacto.Text & "' " & _
            " AND aut_codigo='" & txtAuto.Text & "' "

SQL.Ejecutar(strSQL)
SQL.dataR.Read()
comboMarca.SelectedValue = Valor(SQL.dataR, 1)
comboModelo.SelectedValue = Valor(SQL.dataR, 0)
comboColor.SelectedValue = Valor(SQL.dataR, 2)
txtAno.Text = Valor(SQL.dataR, 3)
txtMotor.Text = Valor(SQL.dataR, 5)
txtChasis.Text = Valor(SQL.dataR, 6)
txtPlaca.Text = Valor(SQL.dataR, 4)
SQL.dataR.Close()
End Sub

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If comprobarDatos() = True Then
        Dim codigoCliente As String = comboCliente.SelectedValue
        Dim codigoAuto As String = comboAuto.SelectedValue
        Dim codigoOrden As String
        Dim codigoProforma As String = ""
        If CBool(comboCliente.Tag) = False Then
            strSQL = " SELECT
cast(CONCAT('C', LPAD(COALESCE(MAX(RIGHT(per_codigo, LENGTH(per_codigo)-
1))+1, 1), 4, '0')) as char) as codigo " & _
                    " FROM persona " & _
                    " WHERE per_que='C' "
            SQL.Ejecutar(strSQL)
            If SQL.dataR.HasRows Then

```

```

        SQL.dataR.Read()
        codigoCliente = SQL.dataR.GetValue(0).ToString
        SQL.dataR.Close()
    End If

    strSQL = " INSERT INTO
persona(per_que,per_codigo,per_tipo,per_ruc," & _
"per_apellido,per_nombre,per_direccion,per_telefono,per_celular," & _
"per_ciudad,ase_nombre) VALUES('C','" & _
codigoCliente & "',''" & comboTipo.SelectedValue &
"'','" & _
Trim(txtRuc.Text) & "',''" & _
Texto(comboCliente.Text) & "',''" & _
Texto(txtNombre.Text) & "',''" & _
Trim(txtDireccion.Text) & "',''" & _
Trim(txtTelefono.Text) & "',''" & _
Trim(txtCelular.Text) & "','0,'" & _
comboAsesor.DisplayMember & "'')'"
    SQL.Ejecutar(strSQL, "L", "I")
End If

If comboAuto.SelectedIndex = 0 Then
    If txtAno.Text = "" Then
        txtAno.Text = "0"
    End If
    strSQL = " INSERT INTO auto(per_codigo,mod_codigo," & _
"aut_ano,aut_placa,col_codigo,aut_chasis,aut_motor) VALUES('" & _
codigoCliente & "',''" & _
comboModelo.SelectedValue & "',''" & _
Texto(txtAno.Text) & "',''" & _
Texto(txtPlaca.Text) & "',''" & _
Texto(txtColor.Text) & "',''" & _
Texto(txtChasis.Text) & "',''" & _
Texto(txtMotor.Text) & "'')'"
    SQL.Ejecutar(strSQL, "L", "I")

    strSQL = " SELECT max(auto_codigo) " & _
" FROM auto " & _
" WHERE per_codigo='" & codigoCliente & "' "
    SQL.Ejecutar(strSQL)
    If SQL.dataR.HasRows Then
        SQL.dataR.Read()
        codigoAuto = Valor(SQL.dataR, 0)
    End If
    SQL.dataR.Close()
End If
If vsfgTipo.Item(0, 0).ToString = "3" Then
    codigoOrden = vsfgTipo.Item(0, 1).ToString

    strSQL = " DELETE FROM orden_trabajo " & _
" WHERE ord_tra_codigo='" & codigoOrden & "' "
    SQL.Ejecutar(strSQL, "L", "E")
    strSQL = " DELETE FROM detalle_orden_trabajo " & _
" WHERE ord_tra_codigo='" & codigoOrden & "' "

```



```

SQL.Ejecutar(strSQL, "L", "E")
ElseIf vsfgTipo.Item(0, 0).ToString = "2" Then
    codigoProforma = vsfgTipo.Item(0, 1).ToString
    strSQL = " UPDATE proforma SET " & _
            " est_prf_codigo=2 " & _
            " WHERE prf_codigo='" & codigoProforma & "' "
SQL.Ejecutar(strSQL, "L", "A")
Dim Num As String = "1"
strSQL = " SELECT
COALESCE(RIGHT(MAX(ord_tra_codigo+1),7),1) AS num " & _
        " FROM orden_trabajo "
SQL.Ejecutar(strSQL)

If SQL.dataR.HasRows Then
    Num = Valor(SQL.dataR, 0)
End If
SQL.dataR.Close()
codigoOrden = Num.PadLeft(7, "0")
Else
Dim Num As String = "1"
strSQL = " SELECT
cast(COALESCE(RIGHT(MAX(ord_tra_codigo+1),7),1) as char) AS num " & _
        " FROM orden_trabajo "

SQL.Ejecutar(strSQL)

If SQL.dataR.HasRows Then
    Num = Valor(SQL.dataR, 0)
End If
SQL.dataR.Close()
codigoOrden = Num.PadLeft(7, "0")
End If
'Ingresa la orden de trabajo
'**** CABECERA
strSQL = " INSERT INTO
orden_trabajo(prf_codigo,ord_tra_siniestro,ord_tra_numero,ord_tra_codigo," & _
"ord_tra_fecha,est_ord_codigo,per_codigo,ord_tra_referencia,ase_codigo," & _
"aut_codigo,ord_tra_perdcto,ord_tra_subtotal,ord_tra_subtotal_o,ord_tra_dct
o," & _
"ord_tra_impuesto,ord_tra_total,ord_tra_observaciones)
VALUES(' " & _
codigoProforma & "','" & Texto(txtSiniestro.Text) &
"',,'" & _
Trim(txtNo.Text) & "','" & _
codigoOrden & "','" & _
Format(dtpFecha.Value, "yyyy-MM-dd") & "','" & _
codigoCliente & "','" & _
Texto(txtReferencia.Text) & "','" & _
comboAsesor.SelectedValue & "','" & _
codigoAuto & "','" & _
NBDD(txtPerDcto.Text) & "','" & _
NBDD(txtSubtotal.Text) & "','" & _

```

```

        NBDD(txtIVA0.Text) & "','" & _
        NBDD(txtDcto.Text) & "','" & _
        NBDD(txtIVA.Text) & "','" & _
        NBDD(txtTotal.Text) & "','" & _
        Texto(txtObservaciones.Text) & "')"
    SQL.Ejecutar(strSQL, "L", "I")

'*****DETALLE
Dim codProducto As String, codGrupo As String
For i As Integer = 1 To vsfgDetalle.Rows.Count - 1
    If vsfgDetalle.Item(i, 1) <> "" Then
        codProducto = vsfgDetalle.Item(i, 9)
        If codProducto = "" Then
            strSQL = " SELECT
COALESCE(MAX(ROUND(pro_codigo))+1,1) FROM producto " & _
                " WHERE pro_tipo='S' "
            SQL.Ejecutar(strSQL)
            SQL.dataR.Read()
            codProducto = CInt(Valor(SQL.dataR, 0))
            SQL.dataR.Close()

            codGrupo = vsfgDetalle.Item(i, 10)
            If codGrupo = "" Then
                strSQL = " SELECT gru_pro_codigo " & _
                    " FROM grupo_producto " & _
                    " ORDER BY gru_pro_codigo LIMIT 1 "
                SQL.Ejecutar(strSQL)
                SQL.dataR.Read()
                codGrupo = Valor(SQL.dataR, 0)
                SQL.dataR.Close()
            End If

            strSQL = " INSERT INTO
producto(pro_tipo,pro_codigo,uni_codigo," & _
                "
pro_nombre,mar_pro_codigo,gru_pro_codigo,pro_costo,pro_iva,pro_descripcion)
VALUES(' " & _
                "'S','" & codProducto & "','" & _
                Texto(vsfgDetalle.Item(i, 1)) & "','" & _
                "'0','"
& codGrupo & "','" & _
                NBDD(vsfgDetalle.Item(i, 2)) & "','" & _
                Math.Abs(CInt(vsfgDetalle.Item(i, 7))) &
                "','" & _
                SQL.Ejecutar(strSQL, "L", "I")

            strSQL = " INSERT INTO
existencia(pro_codigo,exi_cantidad) VALUES(' " & _
                codProducto & "','" & _
                SQL.Ejecutar(strSQL, "L", "I")

        End If

        strSQL = " INSERT INTO
detalle_orden_trabajo(ord_tra_codigo," & _
                "pro_codigo,det_ord_tra_cantidad,det_ord_tra_precio,det_ord_tra_dcto," & _

```

```

"det_ord_tra_dctop,det_ord_tra_total,det_ord_tra_iva) VALUES(' ' & _
        codigoOrden & "',' ' & _
        codProducto & "',' ' & _
        NBDD(vsfsgDetalle.Item(i, 4)) & "',' ' & _
        NBDD(vsfsgDetalle.Item(i, 3)) & "',' ' & _
        NBDD(vsfsgDetalle.Item(i, 5)) & "',' ' & _
        NBDD(vsfsgDetalle.Item(i, 6)) & "',' ' & _
        NBDD(vsfsgDetalle.Item(i, 7)) & "',' ' & _
        Math.Abs(CInt(vsfsgDetalle.Item(i, 8))) & "')"
    SQL.Ejecutar(strSQL, "L", "I")
    End If
Next
MsgBox("Se ha ingresado la Orden de Trabajo No. " &
Trim(txtNo.Text) & " correctamente", MsgBoxStyle.Information, Me.Text)
Limpia()
End If
End Sub

Private Function comprobarDatos() As Boolean
    If Trim(txtNo.Text) = "" Then
        MsgBox("Ingrese el número de orden de trabajo",
MsgBoxStyle.Information, Me.Text)
        txtNo.Focus()
        Return False
    ElseIf comboCliente.Text = "" Then
        MsgBox("Seleccione un cliente", MsgBoxStyle.Information,
Me.Text)
        comboAsesor.Focus()
        Return False
    ElseIf comboAsesor.SelectedIndex <= 0 Then
        MsgBox("Seleccione un asesor del cliente",
MsgBoxStyle.Information, Me.Text)
        comboAsesor.Focus()
        Return False
    ElseIf comboMarca.SelectedIndex <= 0 Then
        MsgBox("Seleccione una marca del auto",
MsgBoxStyle.Information, Me.Text)
        comboMarca.Focus()
        Return False
    ElseIf comboModelo.SelectedIndex <= 0 Then
        MsgBox("Seleccione un modelo del auto",
MsgBoxStyle.Information, Me.Text)
        comboModelo.Focus()
        Return False
    ElseIf vsfgTipo.Item(0, 0).ToString = "1" Then
        Return True
    Else
        Return True
    End If
End Function

Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
    Me.Close()
End Sub

```

```

Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
    Limpiar()
End Sub

Private Sub PonerBotones()

    Dim i As Integer
    Try
        For i = 1 To (vsfgDetalle.Rows.Count - 1)
            vsfgDetalle.Item(i, 0) = i
            vsfgDetalle.SetCellImage(i, 0, ListaImagenes.Images(0))
        Next i
    Catch ex As Exception
    End Try

End Sub

Private Sub vsfgTipo_AfterEdit(ByVal sender As Object, ByVal e As
Cl.Win.ClFlexGrid.RowColEventArgs) Handles vsfgTipo.AfterEdit
    If e.Col = 0 Then
        vsfgTipo.Item(0, 1) = ""
        If vsfgTipo.Item(0, 0).ToString = "2" Then
            'Desde Proforma
            strSQL = " SELECT prf_codigo as
codigo,TRIM(CONCAT(prf_numero,' - ',per_apellido,' ',per_nombre)) as nombre
" & _
                " FROM proforma " & _
                " INNER JOIN persona " & _
                " ON persona.per_codigo=proforma.per_codigo " & _
                " AND persona.per_que='C' " & _
                " WHERE est_prf_codigo=1 " & _
                " ORDER BY prf_numero "
            SQL.Ejecutar(strSQL)
            Dim extral As New Hashtable
            While SQL.dataR.Read
                extral.Add(SQL.dataR.GetValue(0),
SQL.dataR.GetValue(1))
            End While
            SQL.dataR.Close()
            vsfgTipo.Cols(1).DataMap = extral

            lblSiniestro.Visible = True
            txtSiniestro.Enabled = True
            txtSiniestro.Visible = True
        ElseIf vsfgTipo.Item(0, 0).ToString = "3" Then
            'Modificar Orden
            strSQL = " SELECT ord_tra_codigo as
codigo,TRIM(CONCAT(ord_tra_numero,' - ',per_apellido,' ',per_nombre)) as
nombre " & _
                " FROM orden_trabajo " & _
                " INNER JOIN persona " & _
                " ON persona.per_codigo=orden_trabajo.per_codigo "
& _
                " AND persona.per_que='C' " & _

```

```

        " WHERE est_ord_codigo=1 " & _
        " ORDER BY ord_tra_numero "
SQL.Ejecutar(strSQL)
Dim extra As New Hashtable
While SQL.dataR.Read
    extra.Add(SQL.dataR.GetValue(0), SQL.dataR.GetValue(1))
End While
SQL.dataR.Close()
vsfgTipo.Cols(1).DataMap = extra
lblSiniestro.Visible = True
txtSiniestro.Enabled = True
txtSiniestro.Visible = True
Else
    Limpiar()
End If
ElseIf e.Col = 1 Then
    If vsfgTipo.Item(0, 0).ToString = "2" And vsfgTipo.Item(0,
1).ToString <> "" Then
        'Desde Proforma
        ''''CODIGOOO
        Dim num As String = "1"
        strSQL = " SELECT COALESCE(MAX(ROUND(ord_tra_numero))+1,1)
AS num " & _
                " FROM orden_trabajo "
SQLaux.Ejecutar(strSQL)
        If SQLaux.dataR.HasRows Then
            num = Valor(SQLaux.dataR, 0)
        End If
        SQLaux.dataR.Close()
        txtNo.Text = CInt(num)
        Dim observacion As String = ""
        strSQL = " SELECT
prf_codigo,prf_fecha,prf_numero,per_codigo," & _
                "
prf_destinatario,COALESCE(asg_nombre,''),cast(aut_codigo as char)," & _
                " prf_poliza,prf_asegurado,prf_subtotal," & _
                "
prf_subtotal_o,prf_dcto,prf_impuesto,prf_total,prf_observaciones,ase_codigo
,prf_perdcto " & _
                " FROM proforma " & _
                " LEFT JOIN aseguradora " & _
                " ON aseguradora.asg_codigo=proforma.asg_codigo "
& _
                " WHERE prf_codigo='" & vsfgTipo.Item(0,
1).ToString & "' "
SQLaux.Ejecutar(strSQL)
SQLaux.dataR.Read()
        txtAuto.Text = Valor(SQLaux.dataR, 6)
        txtContacto.Text = Valor(SQLaux.dataR, 3)
        comboCliente.SelectedValue = Valor(SQLaux.dataR, 3)
        txtListaPrecio.Text = ht.Item(comboCliente.SelectedValue)

```

```

LimpianNuevo(True, False)
comboAsesor.SelectedValue = Valor(SQLaux.dataR, 15)
comboAuto.SelectedValue = Valor(SQLaux.dataR, 6)
txtSubtotal.Text = Valor(SQLaux.dataR, 9)
txtIVA0.Text = Valor(SQLaux.dataR, 10)
txtDcto.Text = Valor(SQLaux.dataR, 11)
txtIVA.Text = Valor(SQLaux.dataR, 12)
txtTotal.Text = Valor(SQLaux.dataR, 13)
txtPerDcto.Text = Valor(SQLaux.dataR, 16)
observacion = "ASEGURADORA: " & Valor(SQLaux.dataR, 5) & ".
" & IIf(Valor(SQLaux.dataR, 8) = "", "", "ASEGURADO: " &
Valor(SQLaux.dataR, 8) & IIf(Valor(SQLaux.dataR, 7) = "", "", " (" &
Valor(SQLaux.dataR, 7) & ")")) & ". " & Valor(SQLaux.dataR, 14)
txtObservaciones.Text = observacion
txtSiniestro.Text = ""
SQLaux.dataR.Close()
Dim row As Integer = 1
'*** DETALLE
strSQL = " SELECT
pro_codigo,det_prf_precio,det_prf_cantidad, " & _
"
det_prf_dcto,det_prf_dctop,det_prf_total,det_prf_iva,pro_codigo " & _
" FROM detalle_proforma " & _
" WHERE prf_codigo='" & vsfgTipo.Item(0,
1).ToString & "' "
SQLaux.Ejecutar(strSQL)

vsfgDetalle.Rows.Count = 1
While SQLaux.dataR.Read()
vsfgDetalle.Rows.Add()
vsfgDetalle.Item(row, 0) = row
vsfgDetalle.Item(row, 1) =
htProductos(Valor(SQLaux.dataR, 0))
vsfgDetalle.Item(row, 2) =
htGruposProductos(htGrupos(Valor(SQLaux.dataR, 0)))
vsfgDetalle.Item(row, 3) = Valor(SQLaux.dataR, 1)
vsfgDetalle.Item(row, 4) = Valor(SQLaux.dataR, 2)
vsfgDetalle.Item(row, 5) = Valor(SQLaux.dataR, 3)
vsfgDetalle.Item(row, 6) = Valor(SQLaux.dataR, 4)
vsfgDetalle.Item(row, 7) = Valor(SQLaux.dataR, 5)
vsfgDetalle.Item(row, 8) = Valor(SQLaux.dataR, 6)
vsfgDetalle.Item(row, 9) = Valor(SQLaux.dataR, 7)
vsfgDetalle.Item(row, 10) =
htGrupos(Valor(SQLaux.dataR, 0))
row = row + 1
End While
SQLaux.dataR.Close()
txtNo.Focus()

vsfgDetalle.AddItem("")
vsfgDetalle.Item(vsfgDetalle.Rows.Count - 1, 0) =
vsfgDetalle.Rows.Count - 1
vsfgDetalle.SetCellImage(vsfgDetalle.Rows.Count - 1, 0,
ListaImagenes.Imagenes(0))

```

```

        ElseIf vsfgTipo.Item(0, 0).ToString = "3" And vsfgTipo.Item(0,
1).ToString <> "" Then
            'Modificar Orden
            '*** CABECERA
            strSQL = " SELECT
ord_tra_codigo,ord_tra_fecha,ord_tra_numero,per_codigo," & _
                " ord_tra_referencia,ase_codigo,cast(aut_codigo as
char),ord_tra_subtotal," & _
                "
ord_tra_subtotal_o,ord_tra_dcto,ord_tra_impuesto,ord_tra_total,ord_tra_obse
rvaciones,ord_tra_siniestro,ord_tra_perdcto " & _
                " FROM orden_trabajo " & _
                " WHERE ord_tra_codigo='" & vsfgTipo.Item(0,
1).ToString & "' "
            SQLLaux.Ejecutar(strSQL)
            SQLLaux.dataR.Read()
            txtContacto.Text = Valor(SQLLaux.dataR, 3)
            dtpFecha.Value = Valor(SQLLaux.dataR, 1)
            txtNo.Text = Valor(SQLLaux.dataR, 2)
            comboCliente.SelectedValue = Valor(SQLLaux.dataR, 3)
            LimpiarNuevo(True)
            txtReferencia.Text = Valor(SQLLaux.dataR, 4)
            comboAsesor.SelectedValue = Valor(SQLLaux.dataR, 5)
            comboAuto.SelectedValue = Valor(SQLLaux.dataR, 6)
            txtSubtotal.Text = Valor(SQLLaux.dataR, 7)
            txtIVA0.Text = Valor(SQLLaux.dataR, 8)
            txtDcto.Text = Valor(SQLLaux.dataR, 9)
            txtIVA.Text = Valor(SQLLaux.dataR, 10)
            txtTotal.Text = Valor(SQLLaux.dataR, 11)
            txtObservaciones.Text = Valor(SQLLaux.dataR, 12)
            txtSiniestro.Text = Valor(SQLLaux.dataR, 13)
            txtPerDcto.Text = Valor(SQLLaux.dataR, 14)
            SQLLaux.dataR.Close()

            Dim row As Integer = 1
            '*** DETALLE
            strSQL = " SELECT
pro_codigo,det_ord_tra_precio,det_ord_tra_cantidad, " & _
                "
det_ord_tra_dcto,det_ord_tra_dctop,det_ord_tra_total,det_ord_tra_iva,pro_co
digo " & _
                " FROM detalle_orden_trabajo " & _
                " WHERE ord_tra_codigo='" & vsfgTipo.Item(0,
1).ToString & "' "
            SQLLaux.Ejecutar(strSQL)

            vsfgDetalle.Rows.Count = 1
            While SQLLaux.dataR.Read()
                vsfgDetalle.Rows.Add()
                vsfgDetalle.Item(row, 0) = row
                vsfgDetalle.Item(row, 1) =
htProductos(Valor(SQLLaux.dataR, 0))
                vsfgDetalle.Item(row, 2) =
htGruposProductos(htGrupos(Valor(SQLLaux.dataR, 0)))
                vsfgDetalle.Item(row, 3) = Valor(SQLLaux.dataR, 1)
                vsfgDetalle.Item(row, 4) = Valor(SQLLaux.dataR, 2)

```

```

        vsfgDetalle.Item(row, 5) = Valor(SQLaux.dataR, 3)
        vsfgDetalle.Item(row, 6) = Valor(SQLaux.dataR, 4)
        vsfgDetalle.Item(row, 7) = Valor(SQLaux.dataR, 5)
        vsfgDetalle.Item(row, 8) = Valor(SQLaux.dataR, 6)
        vsfgDetalle.Item(row, 9) = Valor(SQLaux.dataR, 7)
        vsfgDetalle.Item(row, 10) =
htGrupos(Valor(SQLaux.dataR, 0))
        row = row + 1
    End While
    SQLaux.dataR.Close()

    vsfgDetalle.AddItem("")
    vsfgDetalle.Item(vsfgDetalle.Rows.Count - 1, 0) =
vsfgDetalle.Rows.Count - 1
    vsfgDetalle.SetCellImage(vsfgDetalle.Rows.Count - 1, 0,
ListaImagenes.Images(0))
        ElseIf vsfgTipo.Item(0, 1).ToString = "" Then
            Exit Sub
        End If
    End If
    PonerBotones()
    calcularTotal()
End Sub

Private Sub vsfgTipo_BeforeEdit(ByVal sender As System.Object, ByVal e
As Cl.Win.C1FlexGrid.RowColEventArgs) Handles vsfgTipo.BeforeEdit
    If e.Col = 1 Then
        If vsfgTipo.Item(0, 0).ToString = "1" Then
            e.Cancel = True
        End If
    End If
End Sub

Private Sub comboTipo_DropDownClosed(ByVal sender As Object, ByVal e As
System.EventArgs) Handles comboTipo.DropDownClosed
    If comboTipo.SelectedValue = "N" Then
        txtNombre.Enabled = True
    Else
        txtNombre.Enabled = False
    End If
End Sub

Private Sub txtNo_LostFocus(ByVal sender As Object, ByVal e As
System.EventArgs) Handles txtNo.LostFocus

    Dim texto() As Char, pos As Integer, cadena As String, cadena2 As
String, cadena3 As Integer
    txtNo.Text = txtNo.Text.ToUpper()
    texto = Aux.ToCharArray
    pos = txtNo.Text.ToUpper().LastIndexOfAny(texto)
    cadena2 = txtNo.Text.Substring(pos + 1)
    cadena3 = CInt(txtNo.Text.Substring(pos + 1) + 1)
    cadena = txtNo.Text.Substring(0, pos + 1) &
txtNo.Text.Substring(pos + 1, Len(cadena2) - Len(cadena3.ToString)) &
cadena3

```



```

Dim Cuantos As Integer = 0
strSQL = " SELECT COUNT(*) " & _
        " FROM orden_trabajo " & _
        " WHERE UCASE(ord_tra_numero)='" & UCase(txtNo.Text) & "' "
SQL.Ejecutar(strSQL)
If SQL.dataR.HasRows Then
    SQL.dataR.Read()
    Cuantos = CInt(SQL.dataR.GetValue(0))
    SQL.dataR.Close()
End If

If Cuantos > 0 And InStr(vsfgTipo.GetDataDisplay(0, 1), txtNo.Text,
CompareMethod.Text) = 0 Then
    ErrorProvider.SetError(txtNo, "El No. de Orden de Trabajo ya
existe, se sugiere utilizar: " & cadena)
Else
    ErrorProvider.SetError(txtNo, "")
End If
End Sub

Private Sub comboColor_SelectedIndexChanged(ByVal sender As
System.Object, ByVal e As System.EventArgs) Handles
comboColor.SelectedIndexChanged
    If comboColor.SelectedValue = "" Then
        txtColor.Text = "0"
    Else
        txtColor.Text = comboColor.SelectedValue
    End If
End Sub

Private Sub txtRuc_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtRuc.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub txtAno_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtAno.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub txtTelefono_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtTelefono.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

Private Sub txtCelular_KeyPress(ByVal sender As Object, ByVal e As
System.Windows.Forms.KeyPressEventArgs) Handles txtCelular.KeyPress
    If InStr(1, "0123456789,-" & Chr(8), e.KeyChar) = 0 Then
        e.KeyChar = ""
    End If
End Sub

End Class

```

Generar Factura

```
Imports System.Collections
Public Class frmFacturacion
    Private strSQL As String
    Private SQL As New clsSQL
    Private SQLaux As New clsSQL
    Private ht As New Hashtable
    Private htCategoria As New Hashtable
    Private htMarcas As New Hashtable
    Private Cargado As Boolean
    Private forma As Form = frmProforma
    Private Sub frmFacturacion_Load(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles MyBase.Load
        SQL.Inicializar(ConnMaster)
        SQLaux.Inicializar(ConnMaster)

        strSQL = " SELECT par_valor as valor " & _
            " FROM parametro " & _
            " WHERE par_codigo='PWD' "
        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        strClavePrincipal = Valor(SQL.dataR, 0)
        SQL.dataR.Close()

        strSQL = " SELECT par_valor as valor " & _
            " FROM parametro " & _
            " WHERE par_codigo='IVA' "
        SQL.Ejecutar(strSQL)
        SQL.dataR.Read()
        strIVA = Valor(SQL.dataR, 0)
        SQL.dataR.Close()
        lblIva.Text = "IVA " & FormatNumber(CInt(strIVA), 0) & "%"
        cargarCombo()
        Limpiar()
    End Sub
    Private Sub cargarCombo()
        Dim ds6 As New DataTable

        strSQL = " SELECT 1,' ' as codigo,'Seleccione una...' as nombre " &
            " UNION " & _
            " SELECT 2,cast(for_pag_codigo as char) as
codigo,for_pag_nombre as nombre " & _
            " FROM forma_pago " & _
            " ORDER BY 1,nombre "
        SQL.Ejecutar(strSQL)
        ds6.Load(SQL.dataR, LoadOption.OverwriteChanges)
        comboFormaPago.DisplayMember = ds6.Columns.Item(2).Caption
        comboFormaPago.ValueMember = ds6.Columns.Item(1).Caption
        comboFormaPago.DataSource = ds6
        SQL.dataR.Close()
    End Sub
End Class
```

```

Private Sub Limpiar()
    Dim num As String = "1"
    txtObservaciones.Text = ""
    strSQL = " SELECT cast(COALESCE(RIGHT(MAX(fac_codigo+1),7),1) as
char) AS num " & _
        " FROM facturacion " & _
        " WHERE fac_tipo='F' "

    SQL.Ejecutar(strSQL)

    If SQL.dataR.HasRows Then
        num = Valor(SQL.dataR, 0)
    End If
    SQL.dataR.Close()
    txtNo.Text = CInt(num)
    comboFormaPago.SelectedIndex = 0
    vsfgDetalle.Rows.Count = 1
    vsfgDetalle.Rows.Count = 2
    Cargar()

End Sub

Private Sub Cargar()
    Dim ds0 As New DataTable

    Dim row As Integer, i As Integer
    strSQL = " SELECT
ord_tra_codigo,ord_tra_numero,ord_tra_fecha,orden_trabajo.per_codigo,CONCAT
(per_apellido,' ',per_nombre) as nombre, " & _
        " TRIM(CONCAT(mar_nombre,' ',mod_nombre,' - ',cast(aut_ano
as char),'
(',aut_placa,')'))),ord_tra_observaciones,COALESCE(CONCAT(asesor.ase_nombre,
' ',ase_apellido),''),ord_tra_siniestro " & _
        " FROM orden_trabajo " & _
        " INNER JOIN persona " & _
        " ON persona.per_codigo=orden_trabajo.per_codigo " & _
        " AND persona.per_que='C' " & _
        " INNER JOIN auto " & _
        " ON orden_trabajo.aut_codigo=auto.aut_codigo " & _
        " AND orden_trabajo.per_codigo=auto.per_codigo " & _
        " INNER JOIN modelo " & _
        " ON modelo.mod_codigo=auto.mod_codigo " & _
        " INNER JOIN marca " & _
        " ON modelo.mar_codigo=marca.mar_codigo " & _
        " LEFT JOIN asesor " & _
        " ON asesor.ase_codigo=orden_trabajo.ase_codigo " & _
        " WHERE est_ord_codigo=1 " & _
        " ORDER BY ord_tra_fecha,ord_tra_numero "

    SQL.Ejecutar(strSQL)
    vsfg.Rows.Count = 1
    row = 1
    While SQL.dataR.Read()
        vsfg.Rows.Add()
        For i = 0 To SQL.dataR.FieldCount - 1

```

```

        vsfg.Item(row, i) = Valor(SQL.dataR, i)
    Next
    row = row + 1
End While
SQL.dataR.Close()

txtSubtotal.Text = FormatNumber(0, 2)
txtIVA.Text = FormatNumber(0, 2)
txtIVA0.Text = FormatNumber(0, 2)
txtTotal.Text = FormatNumber(0, 2)
End Sub

Private Sub CargarDetalle()
    Dim ds0 As New DataTable
    Dim row As Integer, i As Integer, fila As Integer
    fila = vsfg.Row

    txtCodigo.Text = vsfg.Item(fila, 0).ToString
    strSQL = " SELECT
detalle_orden_trabajo.pro_codigo,pro_nombre,det_ord_tra_precio,det_ord_tra_
cantidad, " & _
"
det_ord_tra_dcto,det_ord_tra_dctop,det_ord_tra_total,det_ord_tra_iva " & _
" FROM detalle_orden_trabajo " & _
" INNER JOIN producto " & _
" ON producto.pro_codigo=detalle_orden_trabajo.pro_codigo
" & _
" WHERE ord_tra_codigo='" & vsfg.Item(fila, 0).ToString &
"' "

    SQL.Ejecutar(strSQL)
    vsfgDetalle.Rows.Count = 1
    row = 1
    While SQL.dataR.Read()
        vsfgDetalle.Rows.Add()
        For i = 0 To SQL.dataR.FieldCount - 1
            vsfgDetalle.Item(row, i + 1) = Valor(SQL.dataR, i)
        Next
        row = row + 1
    End While
    SQL.dataR.Close()
    For i = 1 To vsfgDetalle.Rows.Count - 1
        vsfgDetalle.Item(i, 0) = CStr(i)
    Next

    strSQL = " SELECT
ord_tra_subtotal,ord_tra_impuesto,ord_tra_subtotal_o,ord_tra_dcto,ord_tra_t
otal " & _
" FROM orden_trabajo " & _
" WHERE ord_tra_codigo='" & vsfg.Item(fila, 0).ToString &
"' "

    SQL.Ejecutar(strSQL)

    SQL.dataR.Read()
    txtSubtotal.Text = FormatNumber(Valor(SQL.dataR, 0), 2)
    txtIVA.Text = FormatNumber(Valor(SQL.dataR, 1), 2)
    txtIVA0.Text = FormatNumber(Valor(SQL.dataR, 2), 2)

```

```

txtTotal.Text = FormatNumber(Valor(SQL.dataR, 4), 2)
SQL.dataR.Close()
End Sub

Private Sub btnAceptar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnAceptar.Click
    If comprobarDatos() = True Then
        Dim codigoOrden As String = vsfg.Item(vsfg.Row, 0)
        Dim codigoFactura As String = ""

        strSQL = " UPDATE orden_trabajo SET " & _
            " est_ord_codigo=2 " & _
            " WHERE ord_tra_codigo='" & codigoOrden & "' "
        SQL.Ejecutar(strSQL, "L", "A")

        '**** CABECERA
        codigoFactura = txtNo.Text.PadLeft(7, "0")
        strSQL = " INSERT INTO
facturacion(fac_tipo,fac_codigo,ord_tra_codigo,per_codigo,fac_fecha,for_pag
_codigo," & _
            "fac_subtotal,fac_subtotal_o,fac_impuesto,fac_total, "
& _
            "fac_observaciones,fac_serie,fac_numero,fac_anulado)
VALUES('" & _
            "F','" & codigoFactura & "',''" & codigoOrden & "',''" &
vsfg.Item(vsfg.Row, 3) & "',''" & Format(dtpFecha.Value, "yyyy-MM-dd") &
            "',''" & comboFormaPago.SelectedText & "',''" & _
            NBDD(txtSubtotal.Text) & "',''" & _
            NBDD(txtIVA0.Text) & "',''" & _
            NBDD(txtIVA.Text) & "',''" & _
            NBDD(txtTotal.Text) & "',''" & _
            Texto(txtObservaciones.Text) & "',''" & _
            txtSerie.Text & "',''" & _
            CInt(txtNo.Text) & "','0'"
        SQL.Ejecutar(strSQL, "L", "I")

        '*****DETALLE
        For i As Integer = 1 To vsfgDetalle.Rows.Count - 1
            strSQL = " INSERT INTO
detalle_facturacion(fac_tipo,fac_codigo," & _
                "pro_codigo,det_fac_cantidad,det_fac_precio,det_fac_dcto," & _
                "det_fac_dctop,det_fac_total,det_fac_iva) VALUES('" & _
                    "F','" & codigoFactura & "',''" & _
                    vsfgDetalle.Item(i, 1) & "',''" & _
                    NBDD(vsfgDetalle.Item(i, 4)) & "',''" & _
                    NBDD(vsfgDetalle.Item(i, 3)) & "',''" & _
                    NBDD(vsfgDetalle.Item(i, 5)) & "',''" & _
                    NBDD(vsfgDetalle.Item(i, 6)) & "',''" & _
                    NBDD(vsfgDetalle.Item(i, 7)) & "',''" &
                    Math.Abs(CInt(vsfgDetalle.Item(i, 8)))
            & "'')"
            SQL.Ejecutar(strSQL, "L", "I")

```

```

        Next
        MsgBox("Se ha ingresado la Factura No. " & Trim(txtNo.Text) & "
correctamente", MsgBoxStyle.Information, Me.Text)
        Limpiar()

        If MsgBox("Desea imprimir?", MsgBoxStyle.Question +
MsgBoxStyle.YesNo, Me.Text) = MsgBoxResult.Yes Then
            If rptFacturacion.Visible Then
                rptFacturacion.Activate()
            Else
                rptFacturacion.MdiParent = Me
                rptFacturacion.Show()
            End If
        End If
    End If
End Sub

Private Function comprobarDatos() As Boolean
    If Trim(txtSerie.Text) = "" Then
        MsgBox("Ingrese la serie de la factura",
MsgBoxStyle.Information, Me.Text)
        txtNo.Focus()
        Return False
    ElseIf Trim(txtNo.Text) = "" Then
        MsgBox("Ingrese el número de factura", MsgBoxStyle.Information,
Me.Text)
        txtNo.Focus()
        Return False
    ElseIf comboFormaPago.SelectedIndex <= 0 Then
        MsgBox("Seleccione una forma de pago", MsgBoxStyle.Information,
Me.Text)
        comboFormaPago.Focus()
        Return False
    Else
        Dim cod As String, Cuantos As Integer = 0
        cod = txtNo.Text.PadLeft(7, "0")
        strSQL = " SELECT COUNT(*) " & _
                " FROM facturacion " & _
                " WHERE fac_codigo='" & cod & "' " & _
                " AND fac_tipo='F' "
        SQL.Ejecutar(strSQL)
        If SQL.dataR.HasRows Then
            Cuantos = CInt(Valor(SQL.dataR, 0))
        End If
        SQL.dataR.Close()
        If Cuantos > 0 Then
            MsgBox("El número de factura ya está ingresado en el
sistema", MsgBoxStyle.Information, Me.Text)
            txtNo.Focus()
            Return False
        Else
            Return True
        End If
    End If
End Function

```

```

    Private Sub btnCancelar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnCancelar.Click
        Me.Close()
    End Sub

    Private Sub btnLimpiar_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles btnLimpiar.Click
        Limpiar()
    End Sub

    Private Sub PonerBotones()
        Dim i As Integer
        Try
            For i = 1 To (vsfgDetalle.Rows.Count - 1)
                vsfgDetalle.Item(i, 0) = i
                vsfgDetalle.SetCellImage(i, 0, ListaImagenes.Images(0))
            Next i
        Catch ex As Exception
        End Try
    End Sub

    Private Sub vsfg_DoubleClick(ByVal sender As Object, ByVal e As
System.EventArgs) Handles vsfg.DoubleClick
        CargarDetalle()
        Dim num As String = "1"
        txtObservaciones.Text = ""
        comboFormaPago.SelectedIndex = 0
    End Sub

    Private Sub calcularTotal()
        Dim subtotal As Double, dcto As Double, iva12 As Double, iva0 As
Double, total As Double
        For i As Integer = 1 To vsfgDetalle.Rows.Count - 1
            If vsfgDetalle.Item(i, 1) <> Nothing OrElse vsfgDetalle.Item(i,
1) <> "" Then
                If vsfgDetalle.Item(i, 8) = True Then
                    subtotal = subtotal + FormatNumber(vsfgDetalle.Item(i,
7), 2)
                Else
                    iva0 = iva0 + FormatNumber(vsfgDetalle.Item(i, 7), 2)
                End If
                dcto = dcto + FormatNumber(vsfgDetalle.Item(i, 5), 2)
            End If
        Next
        iva12 = FormatNumber(subtotal * 0.12, 2)
        total = FormatNumber(subtotal + iva0 + iva12 - dcto, 2)

        txtSubtotal.Text = FormatNumber(subtotal, 2)
        txtIVA.Text = FormatNumber(iva12, 2)
        txtIVA0.Text = FormatNumber(iva0, 2)
        txtTotal.Text = FormatNumber(total, 2)
    End Sub
End Class

```