

UNIVERSIDAD AMERICANA FACULTAD DE INGENIERIA



DESARROLLO DEL SISTEMA INFORMATICO DE ADMINISTRACIÓN DE VENTAS DE SEGUROS DE LA EMPRESA CORREDORES DE SEGUROS GENERALES S.A

**FELIX OSMIN CONTRERAS BONILLA
RONALD ALEJANDRO ZAMORA CASTILLO**

Monografía para optar al grado de
INGENIERO DE SISTEMAS

Profesor Tutor:
INGENIERO PABLO HURTADO DIAZ

Managua, Nicaragua Abril 2003

DEDICATORIA

Dedicamos este Proyecto Monográfico a nuestros padres, por habernos dado la vida, que con amor, perseverancia y honradez, nos han enseñado el camino a una vida integra y congruente con los principios cristianos.

Por su positividad, apoyo moral y económico pues nos han enseñado a ver lo difícil que es la vida y lo fácil que puede ser cuando ponemos lo mejor de sí en cada acción que ejecutamos.

A nuestros amigos por ayudarnos moralmente, brindándonos aliento y amor para no dejarnos vencer por nada y decirnos siempre la frase "El que quiere puede".

AGRADECIMIENTOS

Queremos agradecer en primer lugar a Dios por habernos dado la sabiduría y el discernimiento necesario para realizar nuestro Proyecto Monográfico.

Agradecer a todas las personas que han puesto su granito de arena en nuestra formación como seres humanos y futuros profesionales sería una lista inmensa difícil de realizar, quizás las principales serían nuestros padres, por su apoyo y motivación incondicional, nuestras familias, por brindarnos el lecho de amor que se necesita en todo momento, nuestros maestros, por transmitirnos la sabiduría necesaria para forjarnos como excelentes profesionales, y a todos nuestros buenos amigos que siempre nos han brindado la oportunidad de escucharnos en nuestros momentos difíciles.

En fin a las personas que nos criticaron a lo largo de todo nuestro camino porque “Solo al árbol que no da frutos no le tiran piedras”.

INDICE GENERAL

PÁG.

INTRODUCCIÓN	
ANTECEDENTES	
JUSTIFICACION	
OBJETIVOS	
Objetivo General	
Objetivos Específicos	
I- MARCO TEORICO	
A- Los Sistemas de Información	6
1- Clasificación de los Sistemas Informáticos	6
1.1- Sistemas Transaccionales	7
1.2- Sistema de Apoyo a la Toma de Decisiones	7
1.3- Sistemas Estratégicos	9
1.4- Sistema de Información para Ejecutivos	9
2- Elementos de los Sistemas informáticos	10
2.1- Base de Datos	10
2.2- Sistema Administrador de Base de Datos	10
2.3- Sistema Operativo	11
2.4- Hardware de Comunicación	11
2.5- Usuarios	12
B- Ciclo de Vida de los Sistema Informáticos	12
1- Modelo en Cascada	12
2- Modelo de Desarrollo Incremental	15
3- Modelo de Prototipo de Requerimientos	15
4- Modelo Espiral	16
5- Modelo Concurrentes	17
6- Selección del modelo de Ciclo de Vida a Utilizar	17
C- Planificación de un Proyecto de Sistema	18
1- Objetivos de la Planificación	18
2- Actividades del Proyecto	19
3- Análisis de Proyectos Informáticos	20
3.1- Identificación de Necesidades	22
3.2- Estudio de Factibilidad	22
3.3- Análisis Económico y Técnico	23
D- Metodologías para el Desarrollo de Sistemas Informáticos	23
1- Metodología Orientada a Datos	23
2- Metodología Orientada a Objetos	24
3- Metodología Orientada a Procesos	24
3.1- Análisis Orientado a Procesos	25
3.1.1- Diagrama de Flujo de Datos (DFD)	25
3.1.1.1- Elementos de un DFD	25
3.1.1.2- Diagrama de Contexto	27
3.1.1.3- Reglas para la Construcción de un DFD	28

3.1.1.4- Regla de Equilibrio	29
3.1.2-Especificaciones de Proceso (PSPEC)	30
3.1.3- Diccionario de Datos	33
3.2- Diseño Orientado a Procesos	33
3.2.1- Descripción General del Sistema	33
3.2.2- Diagrama Entidad / Relación (DER)	33
3.2.2.1- Componentes de un DER	35
3.2.3- Formatos de Entrada y Salida	35
E- Control de Calidad del Software	36
1- Calidad en Ingeniería del Software	36
2- Terminología sobre Calidad	37
3- Fiabilidad del Software	38
3.1- Medidas de Fiabilidad y de Disponibilidad	39
4- Análisis de Riesgo y Seguridad del Software	40
II- MARCO LEGAL	
III- DEFINICIÓN DE LA EMPRESA	
	41
A- ¿Quiénes somos?	41
B- Misión	41
C- Visión	41
D- Nuestros Principios	42
E- Objetivos	42
F- Organigrama	42
G- Sistema Actual	42
IV-ANÁLISIS DEL PROYECTO DE DESARROLLO DEL SISTEMA DE VENTAS DE SEGUROS EN LA EMPRESA CORREDORES DE SEGUROS GENERALES S.A	
	43
A- Alcance	43
B- Definición del Problema	43
1- Descripción del Problema	44
C- Requerimientos	45
D- Metodología	46
E- Actividades	46
F- Estudio de Factibilidad	48
1- Alternativa I	48
1.1- Factibilidad Técnica	49
1.2- Factibilidad Económica	50
1.3- Factibilidad Operativa	50
2- Alternativa II	50
2.1- Factibilidad Técnica	52
2.2- Factibilidad Económica	52

2.3- Factibilidad Operativa	52
G- Análisis Costo – Beneficio	53
1- Punto de Equilibrio	54
2- Análisis Comparativo de la Alternativas	55
3- Dictamen	55
H- Ambiente de Trabajo	
V-ANALISIS ORIENTADO A PROCESOS DEL SISTEMA DE ADMINISTRACIÓN DE VENTAS DE SEGUROS EN LA EMPRESA CORREDORES DE SEGUROS GENERALES S.A	
	56
A-Diagramas de Flujos de datos	56
1- Diagrama de Contexto	57
2- Nivel 1	58
3- Nivel 2	59
4- Nivel 3	60
B- Especificaciones de Procesos	60
1- Nivel 1	63
2- Nivel 2	66
3- Nivel 3	70
C- Diccionario de Datos	70
1- Entidades Externas	71
2- Procesos	75
3- Almacenes de Datos	80
4- Flujos de Datos	
VI-DISEÑO ORIENTADO A PROCESOS DEL SISTEMA DE ADMINISTRACIÓN DE VENTAS DE SEGUROS EN LA EMPRESA CORREDORES DE SEGUROS GENERALES S.A	
	86
A- Descripción del Sistema	88
B- Diagrama Entidad / Relación	88
C- Formatos de Entrada	98
D- Formatos de Salida	
	105
VII-CONCLUSIÓN	
	106
VIII-RECOMENDACIONES	
ANEXOS	
Anexo A- Índice de Tablas	
Anexo B- Índice de Figuras	
Anexo C- Manual de Usuario	

Anexo D- Cotización I
Anexo E- Cotización II
Anexo F- Ley de Derechos de Autor
Anexo G- Código Fuente

BIBLIOGRAFIA

INTRODUCCIÓN

El contenido de la presente tesis es un Proyecto Informático, el cual está dirigido al desarrollo del Sistema de Administración de Ventas de Seguros de la empresa Corredores de Seguros Generales S. A., debido a que, en el transcurso del tiempo, comprendido desde la fundación de la empresa hasta la fecha, la misma ha experimentado un crecimiento organizacional y por lo tanto requiere de un Sistema Informático que llene las necesidades de administración de clientes, control de cuentas por cobrar y facturación, las cuales no son satisfechas hasta este momento.

Nuestra motivación es satisfacer en la medida de lo posible las necesidades antes mencionadas a través del desarrollo de un Sistema Informático escalable que administre las ventas de seguros.

Consientes de que las necesidades antes mencionadas de esta empresa no son satisfechas por los procedimientos manuales empleados en la actualidad, tomamos la ardua tarea de diseñar y desarrollar el Sistema de Administración de Ventas de Seguros.

La presente tesis es un Proyecto Informático que será desarrollado en Microsoft Access 2000, dicho proyecto cuenta con el respaldo de un análisis costo-beneficio.

ANTECEDENTES

Hace aproximadamente tres años, un grupo de personas tomó la decisión de invertir en el campo de los seguros. Estos fundaron una empresa con el nombre **Corredores de Seguros Generales S. A.**, la cual inició operaciones con la venta de una póliza de vida dirigida a los comerciantes de los mercados de Managua.

Desde el inicio, todo el control de la información de los clientes y las operaciones de las pólizas ha sido regido totalmente por procedimientos manuales.

En la etapa inicial el volumen de información era mínimo por lo cual el desorden generado por los procedimientos manuales no era obstáculo para el desarrollo de sus actividades.

Debido a la fuerte demanda de esta póliza, el volumen de clientes ha aumentado considerablemente y con ello el manejo de la información ha sido más difícil de controlar. Por la situación antes descrita, la empresa se ve en la necesidad de controlar y automatizar la Administración de Ventas de Seguros, con el objetivo de brindar un mejor servicio a sus clientes y agilizar las operaciones laborales.

Nicaragua es un país con tecnología de información naciente, en donde no se ha logrado identificar la importancia que los Sistemas Informáticos tienen dentro de una Institución. Según Roger S. Presuman “Hoy en día los Sistemas Informáticos tiene un doble papel. Es un producto y al mismo tiempo el vehículo para hacer la entrega de un producto. Como producto, hace entrega de la potencia informática del hardware informático. Un Sistema Informático es un transformador de información, produciendo, gestionando, adquiriendo, modificando, mostrando o transmitiendo información que puede ser simple o compleja. Como vehículo es utilizado para hacer entrega del producto, el Sistema Informático actúa como la base de control de una computadora, la comunicación de información, y la creación y control de otros programas”¹.

Un Sistema Informático hace entrega de lo que será el producto mas importante de este siglo, la información. Este transforma datos personales para que los datos sean mas

¹ Ingeniería del Software, Un Enfoque Practico, 4ta. Edición, Roger S Pressman, Pág. # 4

útiles en un contexto local, gestiona información comercial para mejorar la competitividad, proporciona el acceso a redes de información por todo el mundo, y también proporciona el medio de adquirir información en todas sus formas.

Debido a lo antes expuesto es sumamente importante desarrollar el Sistema de Administración de Ventas de Seguros, con el objetivo de satisfacer las necesidades surgidas con el paso del tiempo.

JUSTIFICACIÓN

La humanidad se halla en medio de una explosión de la información, un fenómeno tan amplio que esta cambiando el aspecto de toda nuestra sociedad. Hace más de un cuarto de siglo Alvin Toffler, en su famoso libro El shock del futuro, exploró las bases de esta explosión. Entender el origen de la explosión o expansión informativa ayudará a situar en perspectiva la trascendencia de la información para analizar los sistemas de información.

El éxito de toda organización se basa en la agilidad que esta tenga para adaptarse a los cambios, en saber realizar sus actividades de la forma más eficiente, es decir, con los mínimos costos posibles y consiguiendo el máximo rendimiento.

En un mundo tan cambiante como en el que estamos viviendo, una organización que desee superarse no debe quedarse estática ante los cambios del entorno.

En vista de lo antes expuesto, se decidió formular el presente proyecto, el cual es un paso clave para la organización en su desarrollo integral hacia la competitividad y la eficiencia que impone la globalización.

El objetivo final de este proyecto, es satisfacer en la medida de lo posible las necesidades surgidas en el desarrollo de las actividades que a diario se llevan a cabo en la empresa Corredores de Seguros Generales S. A..

La satisfacción de estas necesidades se traducirá en una mejor atención al cliente, información actualizada y disponible en todo momento desde cualquier área de la empresa, mayor seguridad en el acceso a los datos y los procesos operativos y sobre todo se establecerá a la empresa en el nivel competitivo en que se encuentran otras corredurías de seguros.

OBJETIVOS

OBJETIVO GENERAL

Diseñar y Desarrollar el Sistema de Administración de Ventas de Seguros, que incluye el control de Facturación, Cotizaciones, Comisiones por Pagar, Cuentas por Cobrar y Servicios de la Empresa Corredores de Seguros Generales S. A., con el fin de satisfacer sus necesidades, mediante el uso de las herramientas que brindan Microsoft Access 2000.

OBJETIVOS ESPECIFICOS

- Realizar el Análisis del proyecto Desarrollo del Sistema de Administración de Ventas de Seguros de la empresa Corredores de Seguros Generales S. A., mediante la implementación de la teoría Planificación de un Proyecto de Sistemas.
- Cuantificar el valor del proyecto, para conocer el impacto económico que este tendrá en la empresa, mediante un estudio de factibilidad.
- Diseñar el Sistema de acuerdo a las necesidades identificadas del Sistema de Administración de Ventas de Seguros de la empresa Corredores de Seguros Generales S. A., mediante la implementación de la Metodología Orientada a Procesos.
- Desarrollar el Sistema de Administración de Ventas de Seguros de la empresa Corredores de Seguros Generales S. A., con el objetivo de satisfacer sus necesidades, mediante el uso de las herramientas que brindan Microsoft Access 2000.

I- MARCO TEORICO

A- Los Sistemas de Información

“Un Sistema Informático es un conjunto de elementos (Bases de Datos, Programas de Aplicación, Sistema Administrador de Bases de datos (DBMS), Sistema Operativo, Sistema de Comunicación, Hardware y Usuarios) que interactúan entre sí con fin de apoyar las actividades de una empresa”². En un sentido amplio un Sistema Informático no necesariamente incluye hardware. Sin embargo se utiliza como sinónimo de “Sistema de Información Computarizado”.

1- Clasificación de los Sistemas Informáticos

Los sistemas de información cumplen diversos objetivos según los requerimientos de la organización, por esta razón se identifican en base a sus actividades y objetivos a cumplir.

Los Sistemas Informáticos cumplen cuatro objetivos básicos dentro de las organizaciones:

- Automatizar procesos operativos.
- Proporcionar información que sirva de apoyo al proceso de toma de decisiones.
- Lograr ventajas competitivas a través de su implantación y uso.
- Monitorear los factores críticos del éxito del negocio por parte del ejecutivo.

² Folleto de Sistemas Informáticos, Ing. Edgard Robleto

1.1- Sistemas Transaccionales

A través de estos suelen lograrse ahorros significativos de mano de obra, debido a que automatizan tareas operativas de la organización. Se empieza apoyando las tareas en el ámbito operativo de la organización para continuar con los mandos intermedios y, posteriormente con la alta gerencia a medida que evolucionan.

Son fácilmente adaptables a paquetes de aplicación que se encuentran en el mercado, ya que automatizan los procesos básicos que por lo general son similares o iguales en otras instituciones

La integración de éstos sistemas permite que las empresas que lo adquieren tengan el soporte adecuado para utilizar la información que se genera en el proceso de toma de decisiones. Adicionalmente éstos sistemas permiten *exportar* la información, lo cual facilita su uso por otros sistemas desarrollados internamente en las empresas. Esta característica es esencial para que la información se encuentre en una sola base de datos y pueda ser consultada por los diferentes usuarios.

1.2- Sistema de Apoyo a la Toma de Decisiones “DSS” (Decision Support Systems)

“Es un conjunto de programas y herramientas que permiten obtener de manera oportuna la información que se requiere durante el proceso de la toma de decisiones que se desarrolla en un ambiente de incertidumbre”³.

El verdadero objetivo de un sistema de apoyo a las decisiones es proporcionar la mayor cantidad de información relevante en el menor tiempo posible.

Suelen introducirse después de haber implantado los sistemas transacciones más relevantes de la empresa, ya que éstos constituyen su plataforma de información.

No suelen ahorrar mano de obra. Debido a ello, la justificación económica para el desarrollo de éstos sistemas es difícil, ya que no se conocen los ingresos del proyecto de inversión.

³ Folleto de Sistemas Informáticos, Ing. Edgard Robleto

Apoyan a las decisiones que por su misma naturaleza pueden ser repetitivas y estructuradas, así como no repetitivas y no estructuradas.

Los sistemas de soporte a las decisiones *ayudan durante todo el proceso* de la toma de decisiones, no solo para obtener información relevante. En general la información relevante puede provenir de los Sistemas Transaccionales y de los sistemas de información *externos* a la organización.

Los tipos de Sistemas de Apoyo a la Toma de Decisiones son:

- Sistemas de Soporte para la Toma de Decisiones (DSS: Decision Support Systems): Tienen por finalidad apoyar a la toma de decisiones mediante la generación y evaluación sistemática de diferentes alternativas o escenarios de decisión mediante el empleo de modelos y herramientas. Un DSS no soluciona problemas, ya que sólo apoya el proceso de toma de decisiones. La responsabilidad de tomar una decisión es de los administradores.

El DSS puede utilizarse también para obtener información que revele los elementos claves de los problemas y las relaciones entre ellos.

- Sistemas para la Toma de Decisiones de Grupo (GDSS: Group Decision Support Systems): Cumplen el objetivo de lograr participación de un grupo de personas durante la toma de decisiones en ambientes de anonimato y consenso, apoyando decisiones simultáneas.
- Sistemas Expertos de Soporte a la Toma de Decisiones (EDSS: Expert Decision Support Systems): Permiten cargar bases de conocimiento integrados por una serie de reglas de sentido común para que diferentes usuarios las consulten, apoyen la toma de decisiones, la capacitación, etc.

Una de las características que poseen los EDSS es la facilidad de que un usuario, sin tener conocimientos amplios sobre sistemas computacionales, pueda desarrollar sus propios modelos de decisión. Estos modelos son construidos con la ayuda de herramientas de software y hardware.

1.3- Sistemas Estratégicos “SIS” (Strategic Information System)

“Se definen como el uso de la tecnología para apoyar o dar forma a la estrategia competitiva de la organización, a su plan para incrementar o mantener la ventaja competitiva o bien para reducir la ventaja de sus rivales”⁴.

Su función primordial no es apoyar a la automatización de procesos operativos ni proporcionar información para la toma de decisiones. Sin embargo, éste tipo de sistemas puede llevar a cabo dichas funciones.

Su función es lograr ventajas que los competidores no poseen, tales como ventajas en costos y servicios diferenciados con clientes y proveedores. Los sistemas estratégicos son creadores de barreras de entrada al negocio.

Las ventajas que se logran a través de éstos sistemas no son eternas, con el tiempo se convierten en el estándar de la industria.

1.4- Sistema de Información para Ejecutivos “EIS” (Execute Information System)

“Es un sistema computacional que provee al ejecutivo acceso fácil a la información interna y externa al negocio con el fin de dar seguimiento a los factores críticos del éxito”⁵.

De esta definición se desprende el hecho de que los EIS se enfocan principalmente a proporcionar información de la situación actual de la compañía y dejan en un plano secundario la proyección de esta información para escenarios futuros.

El objetivo fundamental de este tipo de sistemas es permitir el monitoreo y seguimiento por parte del ejecutivo de factores críticos del éxito del negocio.

En este punto se definen los factores críticos del éxito como el conjunto de variables de una organización que es necesario monitorear y dar seguimiento para asegurar el éxito de la empresa.

⁴ Folleto de Sistemas de Apoyo a la Toma de Decisiones, Ing. Edgard Robleto

⁵ Folleto de Sistemas de Información para Ejecutivos, Ing. Edgard Robleto

2- Elementos de los Sistemas Informáticos

Como se definió anteriormente un SI es un conjunto de elementos que interactúan entre sí para cumplir un objetivo en común, cada elemento cumple con una función específica que serán descritas a continuación.

2.1- Base de Datos

“Es una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los Sistemas Informáticos de una organización“.

Las bases de datos proporcionan la infraestructura requerida por los sistemas de apoyo a la toma de decisiones y para los sistemas de información estratégicos.

La información recolectada por los sistemas transaccionales o por los sistemas estratégicos es explotada por los sistemas de apoyo a las decisiones o por los mismos sistemas estratégicos.

2.2- Sistema Administrador de Bases de Datos (DBMS)

“Es un software que facilita la comunicación de los usuarios con la Base de Datos, por medio de un lenguaje de consultas y en donde se garantiza la integridad y la seguridad de los datos, así como la recuperación en caso de fallas”⁶.

La función de un DBMS no se limita únicamente a permitir, mediante la definición y manipulación de datos, el diálogo entre los usuarios y la base de datos. Además, debe brindar mecanismos que permitan controlar la concurrencia de varios usuarios que deseen acceder la base de datos en forma simultánea, mantener la seguridad y la integridad de la base de datos, recuperar la base de datos y dejarla en un estado consistente, incluso después de que haya ocurrido una falla del sistema, ya sea provocada por el hardware o por el software.

⁶ Folleto de Bases de Datos y DBMS, Ing. Edgard Robleto

2.3- Sistema Operativo

El Sistema Operativo, es el software que controla todos los recursos de la computadora y establece la base sobre la que pueden escribirse los programas de aplicación. El Sistema Operativo realiza básicamente dos funciones totalmente distintas y es por eso que su definición depende del enfoque de estas dos funciones.

El Sistema Operativo como máquina extendida o máquina virtual: Su deber es “presentar al usuario el equivalente de una máquina extendida o máquina virtual que es más fácil de programar que el hardware subyacente”⁷.

El Sistema Operativo como administrador de recursos: Su deber es “seguir la pista de quién está usando cuál recurso, atender solicitudes de recursos, contabilizar la utilización y mediar entre solicitudes en conflicto provenientes de diferentes programas y usuarios”⁸.

Los elementos básicos en el estudio de cualquier Sistema Operativo son:

- Procesos
- Entrada/Salida
- Administración De La Memoria
- Sistema De Archivos

2.4- Hardware de Comunicación

Son los diferentes dispositivos que intervienen en la conexión de computadoras con el objetivo de compartir el acceso a una computadora anfitriona para ejecutar un programa y procesar datos que se proporcionan desde una terminal distante.

⁷ Sistemas Operativos, Diseño e Implementación, 2da. Edición, Andrew S. Tanenbaum, Pág.# 4

⁸ Sistemas Operativos, Diseño e Implementación, 2da. Edición, Andrew S. Tanenbaum, Pág.# 5

2.5- Usuarios

La necesidad de contar con personal para el desarrollo del software altamente preparado y motivado se viene discutiendo desde los años 60's. "Es un hecho que el factor humano es tan importante que Instituto de Ingeniería de Software ha desarrollado un modelo de madurez de la capacidad de gestión de personal para aumentar la preparación de organizaciones del software para llevar a cabo las cada vez más complicadas aplicaciones"⁹ ayudando a atraer, aumentar, motivar, desplegar y retener el talento necesario para mejorar su capacidad de desarrollo de software.

Se concluye que debido a las características del sistema a desarrollar se puede clasificar como un sistema transaccional porque será un gran recolector de datos y su objetivo principal es automatizar procesos operativos.

B- Ciclo de Vida de los Sistemas Informáticos

Un modelo de ciclo de vida define el estado de las fases a través de las cuales se mueve un proyecto de desarrollo de software.

El primer ciclo de vida del software, "Cascada", fue definido por **Winston Royce** a fines del 70. Desde entonces muchos equipos de desarrollo han seguido este modelo. Sin embargo, ya desde 10 a 15 años atrás, el modelo cascada ha sido sujeto a numerosas críticas, debido a que es restrictivo y rígido, lo cual dificulta el desarrollo de proyectos de software moderno. En su lugar, muchos modelos nuevos de ciclo de vida han sido propuestos, incluyendo modelos que pretenden desarrollar software más rápidamente, o más incrementalmente o de una forma más evolutiva, o precediendo el desarrollo a escala total con algún conjunto de prototipos rápidos.

Existen diversos tipos de modelos entre los principales se encuentran:

1- Modelo en Cascada

Este es el más básico de todos los modelos, y sirve como bloque de construcción para los demás modelos de ciclo de vida. La visión del modelo cascada del desarrollo de software es muy simple; dice que el desarrollo de software puede ser a través

⁹ Ingeniería de Software, Un Enfoque Practico, 4ta. Edición, Roger S. Pressman, Pág. # 39

de una secuencia simple de fases. Cada fase tiene un conjunto de metas bien definidas, y las actividades dentro de una fase contribuye a la satisfacción de metas de esa fase o quizás a una subsecuencia de metas de la fase. Las flechas muestran el flujo de información entre las fases. La flecha de avance muestra el flujo normal. Las flechas hacia atrás representan la retroalimentación.

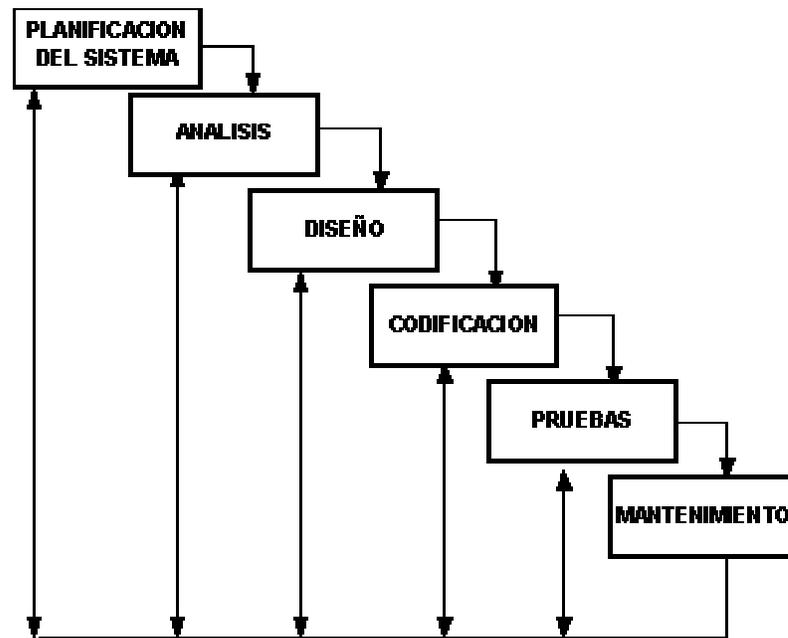


Figura V.1: El Ciclo de Vida Clásico según Roger S. Pressman.

Planificación del Sistema: Es la etapa en la que se determina si el proyecto es o no factible de realizar y se determinan tiempos y costos aproximados, estableciendo así la ruta crítica de cada actividad. Esto es porque la falta de planeación de un sistema es la causa principal de retrasos en programación, incremento de costos, poca calidad, y altos costos de mantenimiento en los desarrollos de productos de software.

Análisis: Es indispensable comprender perfectamente los requisitos del software, para que éste no fracase. Esta etapa puede parecer una tarea relativamente sencilla, pero las apariencias engañan. Abundan los casos en que se puede llegar a malas interpretaciones o falta de información.

Diseño: El diseño del software es realmente un proceso multipaso que se enfoca sobre cuatro atributos distintos del programa: la estructura de los datos, la arquitectura del software, el detalle procedimental y la caracterización de la interfaz. El

proceso de diseño traduce los requisitos en una representación del software que pueda ser establecida de forma que obtenga la calidad requerida antes de que comience la codificación. Al igual que los requisitos, el diseño se documenta y forma parte de la configuración del software.

Codificación: El diseño debe traducirse en una forma legible para la máquina. Si el diseño se realiza de una manera detallada, la codificación puede realizarse mecánicamente.

Prueba: Una vez que se ha generado el código, comienza la prueba del programa. La prueba se centra en la lógica interna del software, asegurando que todas las sentencias se han probado, y en las funciones externas, realizando pruebas que aseguren que la entrada definida produce los resultados que realmente se requieren.

Mantenimiento: Es indudable que el software una vez entregado al cliente sufrirá cambios (posible excepción es el software empotrado). Los cambios ocurrirán debido a que se hayan encontrado errores, a que el software deba adaptarse a posibles cambios.

El desarrollo de productos de software bajo éste ciclo de vida tiene los siguientes problemas:

- Los proyectos reales raramente siguen el flujo secuencial que propone el modelo.
- Normalmente, es difícil para el cliente establecer explícitamente al principio todos los requisitos. Este ciclo lo requiere y tiene dificultades en entender posibles problemas que pudieran existir.
- El cliente debe tener paciencia. Hasta llegar a las etapas finales del desarrollo del proyecto.

A pesar de sus inconvenientes, este ciclo de vida sigue siendo el modelo clásico más ampliamente usado por los ingenieros del software.

2- Modelo de Desarrollo Incremental

Los riesgos asociados con el desarrollo de sistemas largos y complejos son enormes. Una forma de reducir los riesgos es construir sólo una parte del sistema, reservando otros aspectos para niveles posteriores. El desarrollo Incremental es el proceso de construcción siempre incrementando subconjuntos de requerimientos del sistema. Típicamente, un documento de requerimientos es escrito al capturar todos los requerimientos para el sistema completo.

El modelo incremental es 100% compatible con el modelo cascada. El desarrollo incremental no demanda una forma específica de observar el desarrollo de algún otro incremento. Así, el modelo cascada puede ser usado para administrar cada incremento, como se muestra en la figura.

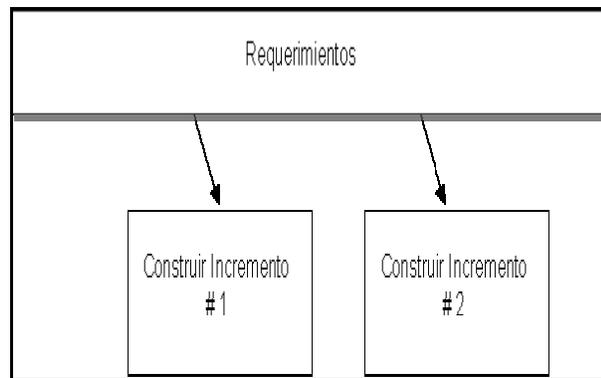


Figura V.2: Modelo de Desarrollo Incremental.

3- Modelo de Prototipo de Requerimientos

El prototipo de requerimientos es la creación de una implementación parcial de un sistema, para el propósito explícito de aprender sobre los requerimientos del sistema. Un prototipo es construido de una manera rápida tal como sea posible. Esto es dado a los usuarios, clientes o representantes de ellos, posibilitando que ellos experimenten con el prototipo.

El Prototipo ha sido usado frecuentemente en los 90, porque la especificación de requerimientos para sistemas complejos tienden a ser relativamente dificultoso de cursar.

Muchos usuarios y clientes encuentran que es mucho más fácil proveer retroalimentación basado en la manipulación, desde un prototipo, en vez de leer una especificación de requerimientos potencialmente ambigua y extensa.

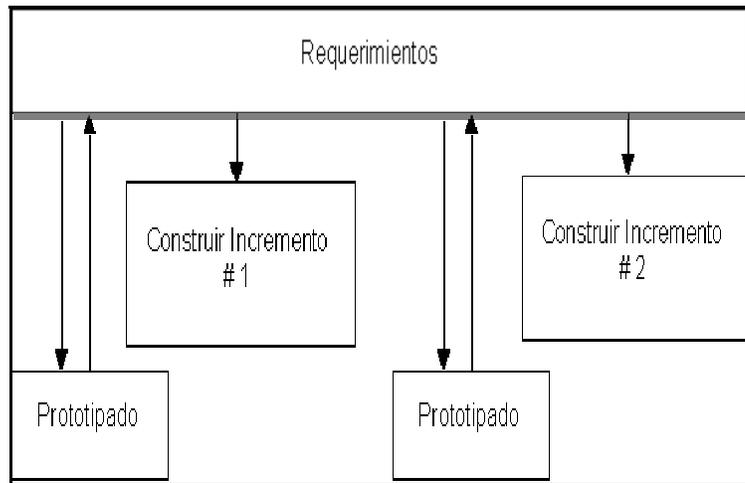


Figura V.3: Prototipo de Requerimientos basado en el modelo de desarrollo incremental

4- Modelo Espiral

El modelo espiral de los procesos software es un modelo del ciclo de meta-vida. En este modelo, el esfuerzo de desarrollo es iterativo. Tan pronto como uno completa un esfuerzo de desarrollo, otro comienza. Además, en cada desarrollo ejecutado, puedes seguir estos cuatros pasos:

- Determinar qué quieres lograr
- Determinar las rutas alternativas que puedes tomar para lograr estas metas. Por cada una, analizar los riesgos y resultados finales, y seleccionar la mejor
- Seguir la alternativa seleccionada en el paso 2
- Establecer qué tienes terminado

5- Modelo Concurrente

Como el modelo espiral, el modelo concurrente provee una meta-descripción del proceso software. Mientras que la contribución primaria del modelo espiral es en realidad que esas actividades del software ocurran repetidamente, la contribución del modelo concurrente es su capacidad de describir las múltiples actividades del software ocurriendo simultáneamente.

Esto no sorprende a nadie que ha estado involucrado con las diversas actividades que ocurren en algún tiempo del proceso de desarrollo de software.

6- Selección del Modelo de Ciclo de Vida a Utilizar

Los modelos presentados anteriormente, suministran una guía con el fin de ordenar las diversas actividades técnicas en el proyecto de desarrollo de software e intentan suministrar un marco para la administración en el desarrollo y el mantenimiento. Aunque todos ellos son compatibles unos con otros, un proyecto puede decidir cuáles enfoques son más útiles en situaciones especiales.

Existe un estudio conducido por **Linda Alexander**, acerca de los factores que influyen en dicha elección. Entre los criterios incluidos en el estudio de Alexander se encuentran:

- Madurez de la aplicación (relacionado a la probabilidad que muchos requerimientos comenzarán a conocerse sólo después del uso del sistema).
- Complejidad del problema y de la solución.
- Utilidad de la funcionalidad rápida (parcial).
- Frecuencias y magnitudes esperadas de los cambios de requerimientos.
- Financiamiento disponible, y su perfil como una función del tiempo.

- Certeza de requerimientos conocidos.

Basado en las características del Sistema de Administración de Ventas de Seguros el modelo que se va a utilizar para su desarrollo es el modelo en Cascada, si bien es cierto en su descripción se habla que presenta grandes críticas, es importante mencionar que este modelo es la base de los demás y se ajusta a nuestras necesidades.

C- Planificación de un Proyecto de Sistemas

“Es el Proceso de gestión para la creación de un Sistema o software, la cual encierra un conjunto de actividades, una de las cuales es la estimación, estimar es echar un vistazo al futuro y aceptamos resignados cierto grado de incertidumbre”¹⁰. Al estimar tomamos en cuenta no solo del procedimiento técnico a utilizar en el proyecto, sino que se toma en cuenta los recursos, costos y planificación.

El tamaño del proyecto es otro factor importante que puede afectar la precisión de las estimaciones. A medida que el tamaño aumenta, crece rápidamente la interdependencia entre varios elementos del Software.

La disponibilidad de información histórica es otro elemento que determina el riesgo de la estimación.

1- Objetivos de la Planificación

Es proporcionar un marco de trabajo que permita al gestor hacer estimaciones razonables de recursos costos y planificación temporal. Estas estimaciones se hacen dentro de un marco de tiempo limitado al comienzo de un proyecto de software, y deberían actualizarse regularmente medida que progresa el proyecto.

El Objetivo de la planificación se logra mediante un proceso de descubrimiento de la información que lleve a estimaciones razonables.

¹⁰ http://www.osmosislatina.com/proyectos_sistemas.htm

2- Actividades del Proyecto

El **Ámbito del Software**, es la primera actividad de llevada a cabo durante la planificación del proyecto de Software.

En esta etapa se deben evaluar la función y el rendimiento que se asignaron al Software durante la Ingeniería del Sistema de Computadora para establecer un ámbito de proyecto que no sea ambiguo, e incomprensible para directivos y técnicos

Describe la función, el rendimiento, las restricciones, las interfaces y la fiabilidad, se evalúan las funciones del ámbito y en algunos casos se refinan para dar mas detalles antes del comienzo de la estimación.

El ámbito se define como un pre-requisito para la estimación y existen algunos elementos que se debe tomar en cuenta como es:

La Obtención de la Información necesaria para el software. Para esto el analista y el cliente se reúnen sobre las expectativas del proyecto y se ponen de acuerdo en los puntos de interés para su desarrollo.

La Segunda tarea de la planificación del desarrollo de Software es la estimación de los recursos requeridos para acometer el esfuerzo de desarrollo de Software, esto simula a una pirámide donde las Herramientas (hardware y Software), son la base proporciona la infraestructura de soporte al esfuerzo de desarrollo, en segundo nivel de la pirámide se encuentran los Componentes reutilizables.

Y en la parte mas alta de la pirámide se encuentra el recurso primario, las personas (el recurso humano).

Cada recurso queda especificado mediante cuatro características:

- Descripción del Recurso.
- Informes de disponibilidad.

- Fecha cronológica en la que se requiere el recurso.
- Tiempo durante el que será aplicado el recurso.

Recursos Humanos: La Cantidad de personas requeridas para el desarrollo de un proyecto de software solo puede ser determinado después de hacer una estimación del esfuerzo de desarrollo (por ejemplo personas mes o personas años), y seleccionar la posición dentro de la organización y la especialidad que desempeñara cada profesional.

Recursos o componentes de software reutilizables: Cualquier estudio sobre recursos de software estaría incompleto sin estudiar la reutilización, esto es la creación y la reutilización de bloques de construcción de Software.

Recursos de entorno: El entorno es donde se apoya el proyecto de Software, llamado a menudo entorno de Ingeniería de Software, incorpora Hardware y Software.

El Hardware proporciona una plataforma con las herramientas (Software) requeridas para producir los productos que son el resultado de la buena practica de la Ingeniería del Software, un planificador de proyectos debe determinar la ventana temporal requerida para el Hardware y el Software, y verificar que estos recursos estén disponibles.

3- Análisis de Proyectos Informáticos

“Es un conjunto o disposición de procedimientos o programas relacionados de manera que juntos forman una sola unidad. Un conjunto de hechos, principios y reglas clasificadas y dispuestas de manera ordenada mostrando un plan lógico en la unión de las partes. Un método, plan o procedimiento de clasificación para hacer algo”¹¹. Esto se lleva a cabo teniendo en cuenta ciertos principios:

- Debe presentarse y entenderse el dominio de la información de un problema.
- Defina las funciones que debe realizar el Software.

¹¹ http://www.osmosislatina.com/proyectos_sistemas.htm

- Represente el comportamiento del software a consecuencias de acontecimientos externos.

- Divida en forma jerárquica los modelos que representan la información, funciones y comportamiento.

El proceso debe partir desde la información esencial hasta el detalle de la Implementación.

La función del Análisis puede ser dar soporte a las actividades de un negocio, o desarrollar un producto que pueda venderse para generar beneficios. Para conseguir este objetivo, un Sistema basado en computadoras hace uso de sus elementos fundamentales.

Un Análisis de Sistema se lleva a cabo teniendo en cuenta los siguientes objetivos en mente:

- Identifique las necesidades del Cliente.
- Evalúe que conceptos tiene el cliente del sistema para establecer su factibilidad.
- Realice un Análisis Técnico y económico.
- Asigne funciones al Hardware, Software, personal, base de datos, y otros elementos del Sistema.
- Establezca las restricciones de presupuestos y planificación temporal.
- Cree una definición del sistema que forme el fundamento de todo el trabajo de Ingeniería.

Para lograr estos objetivos se requiere tener un gran conocimiento y dominio del Hardware y el Software, así como de la Ingeniería humana (Manejo y Administración de personal), y administración de base de datos.

3.1- Identificación de Necesidades

Es el primer paso del análisis del sistema, en este proceso el Analista se reúne con el cliente y/o usuario (un representante institucional, departamental o cliente particular), e identifican las metas globales, se analizan las perspectivas del cliente, sus necesidades y requerimientos, sobre la planificación temporal y presupuestal, líneas de mercadeo y otros puntos que puedan ayudar a la identificación y desarrollo del proyecto.

- Reconocimiento del problema.
- Evaluación y Síntesis.
- Modelado.
- Especificación.
- Revisión.

Antes de su reunión con el analista, el cliente prepara un documento conceptual del proyecto, aunque es recomendable que este se elabore durante la comunicación Cliente – analista, ya que de hacerlo el cliente solo de todas maneras tendría que ser modificado, durante la identificación de las necesidades.

3.2- Estudio de Factibilidad

Muchas veces cuando se emprende el desarrollo de un proyecto de Sistemas los recursos y el tiempo no son realistas para su materialización sin tener pérdidas económicas y frustración profesional. La factibilidad y el análisis de riesgos están relacionados de muchas maneras, si el riesgo del proyecto es alto, la factibilidad de producir software de calidad se reduce, sin embargo se deben tomar en cuenta tres áreas principales de interés:

- Factibilidad Técnica.
- Factibilidad Económica.

- Factibilidad Operativa.

El estudio de la factibilidad puede documentarse como un informe aparte para la alta gerencia.

3.3- Análisis Económico y Técnico

El análisis económico incluye lo que llamamos, el análisis de costos – beneficios, significa una valoración de la inversión económica comparado con los beneficios que se obtendrán en la comercialización y utilidad del producto o sistema.

En el Análisis Técnico, el Analista evalúa los principios técnicos del Sistema y al mismo tiempo recoge información adicional sobre el rendimiento, fiabilidad, características de mantenimiento y productividad.

Los resultados obtenidos del análisis técnico son la base para determinar sobre si continuar o abandonar el proyecto, si hay riesgos de que no funcione, no tenga el rendimiento deseado, o si las piezas no encajan perfectamente unas con otras.

D- Metodologías para el Desarrollo de Sistemas Informáticos

Todos los métodos de análisis de requisitos se basan en la construcción de modelos del sistema que se pretende desarrollar. Utilizando alguna notación, propia de cada método, creamos modelos que reflejen el sistema, y aplicamos técnicas de descomposición y razonamiento top-down, de tal forma que al final establecemos la esencia del sistema que pretendemos construir.

Existen tres tipos de metodologías básicas para el desarrollo de Sistemas Informáticos las cuales son:

1- Metodología Orientada a Datos

El modelo de datos se compone de tres piezas de información interrelacionadas: el objeto de dato, los atributos que describen el objeto de dato, y la relación que conecta los objetos de datos entre sí.

Esta metodología hace uso del Diagrama Entidad / Relación, el cual define todos los datos que se introducen, se almacenan, se transforman y se producen dentro de una aplicación.

2- Metodología Orientada a Objetos

El objetivo de esta es desarrollar una serie de modelos que describan al software de computadora al trabajar para satisfacer un conjunto de requisitos definidos por el cliente.

Para construir un modelo de análisis orientado a objetos se debe tomar en cuenta cinco principios básicos:

- Se modela el dominio de la información
- Se describe la función del modulo
- Se representa el comportamiento del modelo
- Los modelos se dividen para mostrar más detalles
- Los modelos iniciales representan la esencia del problema mientras que los últimos aportan detalles de la implementación

3- Metodología Orientada a Procesos

Esta metodología describe al sistema como un conjunto de operaciones de proceso de información. Estas operaciones reciben unos flujos de datos de entrada y los transforman en flujos de datos de salida. Para describir el sistema desde este punto de vista utilizaremos Diagramas de Flujo de Datos y Especificaciones de Procesos.

Es importante indicar que cualquiera de los tres tipos de diagramas citados van a describir el mismo sistema, que tendrá una implementación física determinada, variando simplemente el punto de vista desde el que se contempla. La relación entre los distintos diagramas se establece a través del Diccionario de Datos.

El análisis y diseño de este sistema se desarrollará mediante la *Metodología Orientada a Procesos*, debido a que este es un sistema transaccional consideramos importante plasmar los procesos que este llevará a cabo.

3.1- Análisis Orientado a Procesos

Este análisis está compuesto por los Diagramas de Flujos de Datos, Especificaciones de Proceso y el Diccionario de Datos.

3.1.1- Diagramas de Flujo de Datos (DFD)

Como su propio nombre indica, un sistema de procesamiento de datos incluye tanto datos como procesos, y cualquier análisis de un sistema así debe incluir ambos aspectos. Necesitamos una técnica para modelar sistemas que describa:

- Qué funciones son las que realiza el sistema
- Qué interacción se produce entre estas funciones
- Qué transformaciones de datos realiza el sistema
- Qué datos de entrada se transforman en qué datos de salida

A medida que la información se mueve a través del software, va siendo modificada mediante una serie de transformaciones. El DFD es una técnica gráfica que representa el flujo de información y las transformaciones que se aplican a los datos al moverse desde la entrada a la salida.

3.1.1.1- Elementos de un DFD

Para representar los elementos de un DFD utilizaremos la notación de Yourdon (1975), posiblemente la más extendida.

Procesos: Representan elementos software que transforman información. Son, por tanto, los componentes software que realizan cada una de las funciones del sistema, transformando datos de entrada en datos de salida.

Entidades externas: Representan elementos del sistema informático o de otros sistemas adyacentes (en cualquier caso se trata de algo que está fuera de los límites del sistema software) que producen información que va a ser transformada por el software o que consumen información transformada por el software. Los flujos de datos que comuniquen el sistema con las entidades externas representan las interfaces del sistema. Las entidades externas sólo aparecen en el diagrama de contexto.

Almacenes de datos: Representan información almacenada que puede ser utilizada por el software. Los almacenes de datos permiten guardar temporalmente información que luego puede ser procesada por el mismo proceso que la creó o por otro distinto. En la mayoría de los casos, utilizaremos almacenes de datos cuando dos procesos intercambian información pero no ocurren o se ejecutan simultáneamente. En otros casos, utilizaremos los almacenes como copia de seguridad de los datos, para evitar pérdidas de información en caso de que el sistema falle. Los almacenes de datos pueden ir desde registros temporales para almacenar un dato hasta ficheros o bases de datos.

Flujos de datos: Representan datos o colecciones de datos que fluyen a través del sistema. La flecha indica el sentido de flujo. Posiblemente en los diagramas de nivel mayor existan flujos de datos bidireccionales, que luego son refinados en sucesivos diagramas, o incluso varios flujos de datos agrupados en uno sólo. Los flujos de datos conectan los procesos con otros procesos, con entidades externas o con almacenes de datos, y pueden converger o divergir si conectan un elemento del DFD con varios otros. Mientras que los almacenes de datos representan información estática o en reposo, los flujos de datos representan información en movimiento. Puede tratarse de un elemento de datos simple o compuesto (un registro) o incluso de una colección de datos de estructura compleja.

Cualquiera de los elementos que aparecen en un DFD tiene que estar etiquetado con un nombre, corto y significativo. Los procesos van etiquetados con la función que realizan, lo mismo sucede con las entidades externas. Los flujos de datos van etiquetados con un nombre identificativo de la información que transportan, y posiblemente con el

estado de dicha información. Los almacenes de datos van etiquetados con un nombre significativo de la información que contienen, generalmente en plural.

Hay que tener en cuenta que un DFD no representa información sobre el comportamiento del sistema o sobre el control del mismo. Representa qué funciones o que transformaciones se realizan sobre los datos pero no cuándo se realizan o en qué secuencia.

3.1.1.2- Diagrama de Contexto

Se pueden utilizar DFDs para representar el sistema a cualquier nivel de abstracción. “El DFD de nivel 0 se llama diagrama de contexto y en él el sistema está representado por un sólo proceso, que identifica cuál es la función principal del sistema, mostrando además los flujos de información que lo relacionan con otros sistemas, las entidades externas”¹². El diagrama de contexto tiene una gran importancia puesto que resume el requisito principal del sistema de recibir ciertas entradas, procesarlas de acuerdo con determinada función y generar ciertas salidas. A partir del diagrama de contexto podemos ir construyendo nuevos diagramas que vayan definiendo con mayor nivel de detalle los flujos de datos y procesos de transformación que ocurren en el sistema, de forma que al final obtenemos una jerarquía de diagramas.

En general, cualquier proceso que aparezca en un DFD puede ser descrito más detalladamente en un nuevo DFD. A esto lo llamaremos explosión de un proceso. En este DFD el proceso que estamos describiendo aparece descompuesto en una serie de subprocesos o subsistemas, cada uno encargado de realizar un aspecto determinado del proceso original. Los flujos de datos que entran y salen del proceso que estamos describiendo deben entrar y salir del DFD que lo desarrolla. Además de estos flujos, el DFD contendrá por lo general nuevos flujos que comunican los procesos que figuran en él y posiblemente almacenes de datos. Las entidades externas sólo aparecen en el DFD de contexto.

Dado que en el DFD de contexto nuestro sistema estará representado por un sólo proceso. Este DFD de nivel 0 sólo dará lugar a un DFD de nivel 1. A su vez, este puede dar lugar a tantos DFDs de nivel 2 como procesos contenga, y así sucesivamente

¹² <http://www.lafacu.com/ análisis estructurado.htm>

hasta que hayamos alcanzado un nivel en el que los procesos sean lo suficientemente simples como para no necesitar su descripción más detallada en un DFD. De esta forma, el modelo de procesos del sistema va a consistir en una jerarquía de DFDs.

3.1.1.3- Reglas para la Construcción de un DFD

- Un DFD debe contener menos de 10 elementos.
- Cada elemento de un DFD debe tener un nombre corto e identificativo.
- Es necesario numerar los procesos.
- Para modelar sistemas complejos se utiliza la explosión, que da como resultado DFDs a distintos niveles de detalle.
- No es conveniente utilizar más de 7 u 8 niveles.
- Los DFDs de niveles inferiores desarrollan de forma más concreta los procesos de niveles superiores.
- La explosión se realiza hasta alcanzar un nivel de especificación mínimo y sencillo.
- Debe mantenerse la consistencia de nombres en los distintos DFDs.
- Debe mantenerse la consistencia entre los distintos niveles, utilizando la regla de equilibrio.
- En cada DFD hijo deben representarse los mismos flujos de datos que en el proceso padre.
- No existen conexiones entre entidades externas.
- No existen conexiones entre entidades externas y almacenes.

- No existen conexiones entre almacenes.

3.1.1.4- Regla de Equilibrio

Normalmente, el sistema que estamos modelando será lo suficientemente complejo como para necesitar varios DFDs. Construiremos entonces una jerarquía de DFDs.

Cada DFD (hijo) de un nivel n será resultado de la explosión de un proceso (padre) de un DFD de nivel $n-1$. Es necesario que el título del DFD sea el nombre del proceso que desarrolla, que la numeración de los procesos en el DFD hijo se derive del número del DFD padre (por ejemplo 3, 3.1, 3.2) y además hay que mantener la consistencia de los flujos de datos en ambos.

La regla de equilibrio dice que los flujos de datos que entran o salen del proceso padre deben aparecer en el DFD hijo, manteniendo el nombre y el sentido.

En el diagrama hijo estos flujos de datos tendrá un extremo libre, puesto que conectaban el proceso padre con algún elemento que ya no va a estar representado en el diagrama hijo. Una excepción son los flujos que conectan los procesos con almacenes de datos. Se recomienda representar los almacenes de datos en todos los DFDs en los que intervienen.

La regla de equilibrio también puede tener excepciones. Antes habíamos hablado de flujos de datos compuestos, e incluso bidireccionales (que normalmente transmiten una pregunta o petición y su respuesta). Estos flujos de datos suelen utilizarse en los DFDs de los niveles más bajos (es decir, los más altos de la jerarquía) para modelar el sistema con cierto grado de abstracción y para evitar representar muchos flujos de datos, lo que haría más confuso el diagrama. En los DFDs de niveles superiores, es necesario descomponer estos flujos en varios (por ejemplo. Petición y respuesta). Para realizar esta descomposición podemos hacer divergir el flujo en el DFD hijo o al menos dejar reflejada esta descomposición en el diccionario de datos.

3.1.2- Especificaciones de Proceso (PSPEC)

Los DFDs permiten identificar las principales funciones o transformaciones que realiza un sistema, y las relaciones entre estas funciones pero no indican nada acerca de los detalles de cómo se realizan estas transformaciones. Para definir los detalles de qué información de entrada se transforma en qué información de salida y cómo se realiza esta información se necesita una descripción textual de los procesos. Para esto utilizaremos las especificaciones de proceso.

En los DFDs de menor nivel (los más altos en la jerarquía), los procesos se describen mediante un nuevo DFD, que define más detalladamente las funciones que realiza y los flujos que maneja. Este proceso de descomposición debe continuar hasta que se alcance un nivel en el que un proceso puede ser descrito textualmente de forma sencilla y no ambigua. Estos procesos de los nodos hoja de la jerarquía se suelen llamar primitivas de proceso.

Una forma de especificar una primitiva de proceso es dando un algoritmo. Esto es simplemente una forma de describir la operación.

Alternativamente, una primitiva de proceso puede ser descrita mediante una definición, sin especificar un algoritmo, matemáticamente o mediante un lenguaje de especificación formal, en lenguaje natural o mediante una tabla que indica los valores de los flujos de salida a partir de los valores de los flujos de entrada.

En cualquier caso, las especificaciones de proceso son textos breves, que definen la función que realiza la primitiva de proceso para transformar los flujos de entrada en flujos de salida.

3.1.3- Diccionario de Datos

“El diccionario de datos o diccionario de requisitos es el lugar donde van a estar contenidas las definiciones de todos los elementos que aparecen en los distintos diagramas del sistema, y sirve por tanto para establecer la relación entre los distintos modelos del mismo (procesos, comportamiento y datos). El diccionario de datos permite establecer que

el analista y el usuario entienden de la misma forma cada uno de los elementos de los diagramas”¹³.

El análisis del sistema software no va a estar completo porque hagamos una serie de diagramas DFD, DFC, DER y DE. Necesitamos una cierta cantidad de información textual, que defina brevemente cada uno de los elementos de los modelos (con las etiquetas no basta), o que describa el contenido de dichos elementos (el contenido de los flujos y almacenes, de entidades y relaciones, el pseudo código de los elementos de proceso).

La información que suele contener el diccionario de datos es la siguiente:

Nombre: El nombre principal del elemento en el sistema.

Alias: Otros nombres usados para el mismo elemento en diferentes partes del sistema. Aunque lo ideal sería que un mismo elemento se llamase igual en todos los modelos, hay que tener en cuenta que puede ser que la labor de análisis no sea hecha por una sola persona sino por un equipo de analistas, y en estos casos la disparidad de nombres suele ser inevitable.

Uso: Un listado de dónde (qué procesos o diagramas) se usa el elemento y cómo se usa (entrada, salida, actualización). Esta información se puede generar automáticamente con una herramienta de análisis o una herramienta CASE y es muy útil a la hora de estimar el impacto de un cambio (sí queremos modificar el tipo de un dato elemental, cuántos programas es necesario modificar).

Descripción: Una descripción breve del significado del elemento, que complete la información del nombre.

Contenido: Una relación de los datos elementales que contiene, en el caso de ser un flujo, un almacén, un objeto o una relación. Para la descripción del contenido de los elementos existe una notación normalizada.

¹³ http://www.lafacu.com/analisis_estructurado.htm

Información adicional: Otras informaciones relativas al elemento (quién y cuándo lo definieron) o información específica del tipo de elemento de que se trate.

En concreto, un diccionario de requisitos completo debe incluir entradas para los siguientes elementos:

- Entidades externas: Descripción.
- Procesos: Descripción de la función que realizan.
- Flujos de datos: Descripción, dónde se usan y su contenido.
- Almacenes de datos: Descripción, dónde se usan y contenido.
- Datos elementales: Descripción, tipo y restricciones (p.ej. de rango o de valores)
- Primitivas de proceso: Descripción y pseudo código.
- Flujos de control: Descripción, dónde se usan y su contenido.
- Almacenes de control: Descripción, dónde se usan y contenido.
- Estados: Descripción.
- Transiciones: Condiciones de activación y acciones a realizar (sí es necesario).
- Entidades: Descripción y atributos que contienen.
- Relaciones: Descripción y atributos que contienen.

Actualmente, el diccionario de datos suele estar mecanizado, bien sea por separado o bien como parte de una herramienta CASE. Esto hace que las modificaciones sean más fáciles, que las comprobaciones de que cada modelo o el conjunto de los modelos

son correctos se puedan hacer de forma automática o que sirva de ayuda para evaluar el impacto de un cambio en el sistema.

3.2- Diseño Orientado a Procesos

Este diseño esta compuesto por la Descripción General del Sistema, Diagrama Entidad / Relación (DER), Formatos de Entrada y Formatos de Salida.

3.2.1- Descripción General del Sistema

Apoyan el proceso de formular las características que debe tener una aplicación, tales como Entradas, Salidas, Procesamiento y Especificaciones de Control. Muchas incluyen herramientas para crear especificaciones de datos.

Se utilizan para describir la posición de datos, mensajes y encabezados sobre las pantallas de las terminales, reportes y otros medios de entrada y salida.

3.2.2- Diagrama Entidad / Relación (DER)

El Diagrama Entidad / Relación se centra únicamente en los datos que maneja el proceso y en qué relaciones se establecen entre estos datos, dejando de lado las funciones que realiza el sistema y el control del mismo (Esta abstracción era una de las características de los modelos).

3.2.2.1- Componentes de un DER

Objetos (o tipos de objetos): Representados mediante un rectángulo etiquetado. Modelan una colección o conjunto de objetos o cosas del mundo real, cuyos miembros individuales (instancias) juegan un papel necesario en el sistema, pueden ser identificados individualmente (mediante alguna clave o identificador) y pueden ser descritos mediante una serie de datos elementales (atributos).

Relaciones: Los objetos de un DER se conectan unos a otros mediante relaciones, representadas mediante un rombo etiquetado. Cada instancia de una relación

representa una asociación entre cero o más instancias de cada uno de los objetos conectados a ella. En algunos casos la relación asocia diferentes instancias del mismo objeto.

Podemos definir la cardinalidad de la relación etiquetando los conectores con números o rangos (0:N). De esta forma indicamos cuántas instancias de cada objeto intervienen en la relación.

Hay que tener en cuenta que las relaciones representan algo que tiene que ser recordado por el sistema, es decir, que no puede ser calculado o derivado automáticamente, sino cada instancia de una relación, que asocia instancias de los diferentes objetos tiene que estar registrada o almacenada de alguna forma en el sistema. Si una relación entre varios objetos puede ser calculada de alguna forma, no la representaremos en el diagrama.

Otro detalle es que las relaciones pueden asociar instancias de múltiples objetos, no sólo dos. Normalmente, la relación puede ser descrita desde el punto de vista de cada uno de los objetos asociados a ella. En otros casos, habrá un objeto que determine el punto de vista desde el que se debe leer la relación, en estas situaciones podemos asignar un sentido a los conectores.

Por último, en un DER no indicaremos todas las relaciones que existen entre los objetos, sino sólo aquéllas que sean relevantes para el sistema.

Atributos: Los atributos son datos elementales asociados a un objeto o a una relación. Podemos representarlos mediante elipses que cuelguen del objeto o relación o no representarlos en el modelo, detallándolos entonces en la entrada del diccionario de datos dedicada al objeto o relación en donde están incluidos.

Indicadores de supertipo y subtipo: Podemos establecer taxonomías entre los diferentes objetos relevantes para el sistema. Conectaremos estos objetos de forma jerárquica. Los objetos de los subtipos contendrán atributos comunes (que colgaremos del supertipo) y otros específicos, que colgaremos de cada subtipo particular.

3.2.3- Formatos de Entrada y Salida

Los formatos de entrada describen como se comunica el software consigo mismo, con los sistemas que operan junto con él y con los operadores y usuarios que lo emplean.

Los formatos de salida se refieren a los resultados e informaciones generadas por el Sistema, Para la mayoría de los usuarios la salida es la única razón para el desarrollo de un Sistema y la base de evaluación de su utilidad.

E- Control de Calidad del Software

La definición oficial, y más completa, es la de la norma ISO 8402 (admitida como norma española UNE 66-001-92), que define la calidad, en general, como: “Totalidad de características de un producto o servicio que le confieren su aptitud para satisfacer unas necesidades expresadas o implícitas”¹⁴.

Esta definición nos permite comprender que la consecución de la calidad puede tener tres orígenes:

La calidad realizada: La que es capaz de obtener la persona que realiza el trabajo. El ejemplo más típico es el programador que trabaja sin especificaciones, pero se refiere también al grado de cumplimiento de la especificación que el responsable de un trabajo es capaz de conseguir.

La calidad programada: La que se ha pretendido obtener. Es la que aparece descrita en un documento de diseño, en una especificación o en un plano constructivo. Es, por tanto, la que se le ha encomendado conseguir al responsable de ejecutar el trabajo.

La calidad necesaria: La que el cliente exige con mayor o menor grado de concreción o, al menos, la que le gustaría recibir.

¹⁴ Análisis y Diseño Detallado de Aplicaciones, Mario G. Piattini, Joaquín Cervera, Luis Fernández, Pág. # 447

La gestión de la calidad pretenderá conseguir que estos tres círculos coincidan entre sí. Todo lo que se encuentre fuera de dicha coincidencia será motivo de derroche, de gasto superfluo o de insatisfacción.

1- Calidad en Ingeniería del Software

Lo primero que se debe considerar a la hora de abordar el tema de la calidad del software es que este constituye un producto con unas características muy peculiares, como son:

- Se desarrolla, no se fabrica en el sentido clásico del término.
- Se trata de un producto lógico, sin existencia física.
- No se degrada con el uso.
- Se entrega conscientemente con defectos.

Debido a estas características, se dice que la calidad del software se debe obtener a medida que se construye; no es un añadido que podamos poner una vez desarrollado el software.

La definición oficial de calidad del Software es la del estándar IEEE std.610-1990, la cual es:

“Grado con el que un sistema, componente o proceso cumple, los requisitos especificados o las necesidades o expectativas del cliente o usuarios”¹⁵.

2- Terminología sobre Calidad

Para poder afrontar el estudio de la calidad del software debemos conocer primero los principales términos empleados en esta área, como son:

¹⁵ Análisis y Diseño Detallado de Aplicaciones, Mario G. Piattini, Joaquín Cervera, Luis Fernández, Pág. # 450

Gestión de la calidad del software: Aspecto de la función general de la gestión que determina y aplica la política de calidad. Normalmente, la gestión de calidad se aplica a nivel de empresa, por lo que incluye planificación estratégica, asignación de recursos, etc.

Aseguramiento de la calidad del Software: Conjunto de actividades planificadas y sistemáticas necesarias para aportar la confianza en que el producto cumplirá los requisitos dados de calidad.

Control de calidad del software: Técnicas y actividades de carácter operativo, utilizadas para satisfacer los requisitos relativos a la calidad, centradas en dos objetivos fundamentales: mantener bajo control un proceso y eliminar las causas de defectos en las diferentes fases del ciclo de vida.

Verificación y validación del software: Es una actividad ligada al control de calidad en el ámbito del software.

Verificación: En general, se trata de comprobar si los productos construidos en una fase del ciclo de vida satisfacen los requisitos establecidos en la fase anterior.

Validación: En general, se trata de comprobar si el Software construido satisface los requisitos del usuario.

3- Fiabilidad del Software

No hay duda de que la fiabilidad de un programa de computadora es un elemento importante de su calidad general. Si un programa falla frecuentemente y repetidamente en su funcionamiento, no importa si el resto de los factores de calidad son aceptables.

La fiabilidad del software se define en términos estadísticos como “La probabilidad de operación libre de fallos de un programa de computadora en un entorno determinado y durante un tiempo específico”¹⁶.

¿Qué se entiende por el término fallo? En el contexto de cualquier discusión sobre calidad y fiabilidad del software, el fallo es cualquier falta de concordancia con los requisitos del software.

3.1- Medidas de Fiabilidad y de Disponibilidad

En el hardware, son más probables los fallos debido al desgaste físico que los fallos relativos al diseño. Desgraciadamente, para el software lo que ocurre es lo contrario, de hecho, todos los fallos del software, se producen por problemas de diseño o de implementación.

Considerando un sistema basado en computadora, una sencilla medida de la fiabilidad es el tiempo medio entre fallos (TMEF), donde:

$$\text{TMEF} = \text{TMDF} + \text{TMDR}$$

Las siglas TMDF y TMDR corresponden al tiempo medio de fallo y al tiempo medio de reparación, respectivamente.

Además de una medida de la fiabilidad debemos obtener una medida de la disponibilidad. La disponibilidad del software es la probabilidad de que un programa funcione de acuerdo con los requisitos en un momento dado, y se define como:

$$\text{Disponibilidad} = \text{TMDF} / (\text{TMDF} + \text{TMDR}) \times 100\%$$

La medida de fiabilidad TMEF es igualmente sensible al TMDF que al TMDR. La medida de disponibilidad es algo más sensible al TMDR, una medida indirecta de la facilidad de mantenimiento del software.

¹⁶ Ingeniería de Software, Un Enfoque Practico, 4ta. Edición, Roger S. Pressman, Pág. # 132

4- Análisis de Riesgo y Seguridad del Software

El análisis de riesgo y la seguridad del software son actividades de garantía de calidad del software que “se centra en la identificación y evaluación de los riesgos potenciales que pueden producir un impacto negativo en el software y hacer que falle el sistema completo”¹⁷. Si se pueden identificar pronto los riesgos en el proceso de ingeniería del software podrán especificarse las características del diseño del software que permitan eliminar o controlar los riesgos potenciales.

Como parte de la seguridad del software, se puede dirigir un proceso de análisis y modelado. Inicialmente, se identifican los riesgos y se clasifican por su importancia y su grado de riesgo.

Cuando se han identificado los riesgos del sistema, se utilizan técnicas de análisis para asignar su gravedad y su probabilidad de ocurrencia. Para que sea efectivo, se tiene que analizar el software en el contexto del sistema completo.

Cuando se han identificado y analizado los riesgos, se pueden identificar requisitos del software relacionados con la seguridad.

Aunque la fiabilidad y seguridad del software están bastante relacionadas, es importante entender la sutil diferencia que existe entre ellas. La fiabilidad del software utiliza el análisis estadístico para determinar la probabilidad de que pueda ocurrir un fallo del software. Sin embargo, la ocurrencia de un fallo no lleva necesariamente a un riesgo o a un accidente. La seguridad del software examina los modos según los cuales los fallos producen condiciones que pueden llevar a accidentes.

¹⁷ Ingeniería de Software, Un Enfoque Practico, 4ta. Edición, Roger S. Pressman, Pág. # 133

II- MARCO LEGAL

El estudio de un Marco Legal pretende, señalar y definir claramente los conceptos legales, externo a la institución, los cuales afectan diversas decisiones entorno al desarrollo del proyecto. Específicamente se va a considerar la ley de Derechos de Autor recientemente aprobada por la asamblea Nacional y publicada en la Gaceta el 31 de Agosto de 1999.

Es importante señalar que esta ley afecta directamente al proyecto ya que este requiere de recursos informáticos de software como parte integrante del proceso de automatización, los cuales están protegidos por la ley de Derechos de Autor y derechos de Propiedad.

Además, debemos identificar las acciones a tomar para la adquisición y legalización de productos informáticos protegidos por la ley de Derechos de Autor para no incurrir en acciones que puedan ser consideradas ilegales o que al menos estén sujetas a dudas.

También debemos señalar que las ideas utilizadas en el desarrollo de este sistema son originales y se ha llegado a ellas a través de la investigación en diferentes medios y entrevistas con personas expertas en la materia. Por este motivo la propiedad intelectual del sistema y los derechos de autor pertenecen a los desarrolladores del mismo y a la Universidad Americana. Ninguna modificación podrá ser aplicada al sistema sin el consentimiento expreso a los desarrolladores. El sistema no puede ser comercializado unilateralmente por la Universidad Americana ya que su naturaleza es académica y científica para la sociedad nicaragüense.

Para mayor información ver Anexo F

III- DEFINICION DE LA EMPRESA

A- ¿Quiénes somos?

Somos un Correduría de Seguros con capital 100% Nicaragüense con más de 3 años de experiencia en el mercado asegurador, fundado el 24 de Febrero de 2000. Desde el principio de nuestra operación como intermediarios de seguros, hemos ocupado un lugar predominante dentro de la industria aseguradora en Nicaragua, manteniéndonos siempre en las primeras posiciones en participación del mercado.

B- Misión

Somos una de Correduría de Seguros, orientamos nuestro esfuerzo creativo y profesional a la identificación y satisfacción de las necesidades de aseguramiento de las personas, sus familias, las instituciones y las empresas; generando empleo y bienestar a todos nuestros funcionarios.

C- Visión

Posicionarnos como el intermediario de seguros más ágil en la atención de las necesidades de nuestros clientes mediante soluciones novedosas y efectivas. Destacarnos por la prestación de servicios de excelencia, llegando a ser uno de los Corredores de Seguros más importantes en el país.

D- Nuestros Principios

Inculcamos a nuestro grupo humano el trabajar bajo estos principios:

- Calidad
- Respeto
- Honestidad
- Prudencia
- Ética
- Confidencialidad
- Equidad
- Eficiencia

E- Objetivos

- Dar un seguimiento exhaustivo del mercado asegurador, para disponer de los mejores precios.
- Capacitar continuamente a nuestro personal para que pueda dar la mejor respuesta en cada momento a las necesidades de nuestra clientela.

F- Organigrama

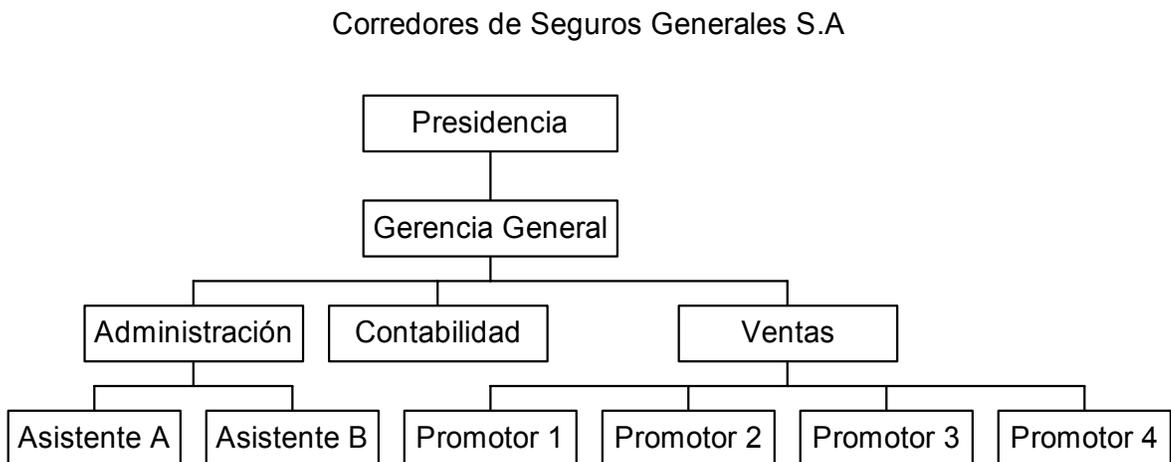


Figura VII.1: Organigrama de la Empresa

G- Sistema Actual

En la actualidad Corredores de Seguros Generales no cuenta con ningún tipo de Sistema de Información, lo que hace que el Sistema de Administración de Ventas de Seguros sea el pionero en este ámbito

IV- ANÁLISIS DEL PROYECTO DE DESARROLLO DEL SISTEMA DE ADMINISTRACIÓN DE VENTAS DE SEGUROS EN LA EMPRESA CORREDORES DE SEGUROS GENERALES S.A

A- Alcance

Esta aplicación tiene como alcance todo lo relacionado con la administración de ventas de seguros y el pago de comisiones a los promotores de la empresa, el cual no tiene ninguna relación con el módulo de contabilidad.

Esta aplicación recolectará los datos de mayor importancia sobre las pólizas y los clientes así como también sobre las ventas efectuadas, a partir de los cuales la aplicación producirá informes y gráficos estadísticos definidos por el usuario para una mejor visión de lo que ocurre en la empresa.

B- Definición del Problema

La empresa Corredores de Seguros Generales, tiene como principal problema la desorganización de información, la cual no le permite tener una mejor atención hacia sus clientes, incursionar con nuevos productos, maximizar las horas laborales de sus empleados, cumplir con el estándar tecnológico de las demás corredurías de seguro, entre otras desventajas.

1- Descripción del Problema

Corredores de Seguros Generales S. A. es una compañía situada en Ciudad Jardín que se dedica a la venta de Seguros al contado o al crédito. Ofrecen una amplia gama de pólizas las cuales van desde un seguro de automóvil hasta un seguro de gastos médicos o de vida. En la actualidad Corredores de Seguros Generales S. A., no cuenta con ningún tipo de Sistemas de información por lo tanto es administrada manualmente. El registro manual de cada cliente y venta es efectuado por el promotor que la atendió sin ningún tipo de orden, lo cual lo hace aun más desorganizado.

Al no tener un Sistema de Información, los promotores no tienen una guía actualizada y confiable de los clientes que se deben visitar tanto para su renovación de

pólizas como para el cobro de las mismas. Otra debilidad que presenta al no tener un Sistema es la poca credibilidad que se le demuestra al cliente a la hora de presentar sus avisos de cobro, debido que lo que usan actualmente son efectuados a mano por el promotor y sin ningún historial de los pagos efectuados anteriormente.

Estas debilidades mencionadas anteriormente son las más urgentes de solucionar, pero sin embargo la implementación de este Sistema de Administración de Ventas de Seguros no se detiene en la solución de estas debilidades, al brindar una amplia gama de facilidades para la empresa lo que le da un valor agregado al Sistema.

C- Requerimientos

Los requerimientos por lo que será diseñado el Sistema de Administración de Ventas de Seguros se dividen en tres categorías, entrada, salidas y peticiones:

Entradas al sistema:

- Ingreso de datos de las Pólizas
- Ingreso de datos de los Promotores
- Ingreso de datos de los Clientes
- Ingreso de Ventas

Salidas del sistema:

- Emisión de Avisos de Cobros por Ventas
- Emisión de Facturas
- Pago de Comisiones
- Elaboración de Cotizaciones

Peticiones del usuario:

- Estados de Cuenta por Ventas y por Clientes
- Reimpresión de Facturas
- Análisis de Ventas
- Informe de Moras
- Búsqueda y Cancelación de Facturas

No se requiere compartir datos por lo que el personal del departamento es pequeño (dos personas).

D- Metodología

Para modelar este sistema de administración de Ventas de Seguros se usará la metodología de análisis y diseño estructurados Orientada a Proceso.

Se implementará el ciclo de vida clásico debido que involucra al usuario final en la detección y corrección de errores desde el principio del proyecto. Esto ayudará a sanear previamente cualquier mal funcionamiento que pueda presentarse.”

E- Actividades

Tabla VIII.1: Actividades a realizar en el desarrollo de la Aplicación

Núm.	Actividad	Duración (semanas)
A	Selección, definición y descripción del problema, recolección de datos de interés, elaboración de objetivos	2
B	Realizar el estudio de Factibilidad	1
C	Realizar el análisis Costo – Beneficio	1
D	Desarrollo de la Aplicación	4
E	Prueba y comprobación	1
F	Elaboración de la documentación	2
G	Capacitación de personal	2
Total		13

F- Estudio de Factibilidad

Este estudio de Factibilidad se ha elaborado para determinar que tan viable es la implementación de un sistema informático que administre las ventas de seguros y que satisfaga las necesidades informáticas de la empresa. Este estudio presenta dos distintas opciones para dar respuesta a las necesidades que afronta actualmente dicha empresa. Estas dos alternativas surgieron como respuesta a la situación y el contexto en que se maneja la información. Con las distintas técnicas y métodos de análisis de la ingeniería del software se tiene la capacidad de presentar alternativas de software de una alta confiabilidad, y de emitir un criterio acerca de estas propuestas, lo que permitirá tomar una decisión adecuada a las posibilidades y recursos disponibles.

Este sistema de administración de ventas de seguro va a permitir llevar un control automatizado y actualizado de los datos de las ventas, clientes y pólizas, así como también le permitirá a la empresa configurar o adaptar el sistema a sus políticas internas de la institución, estas configuraciones van desde la cantidad de beneficiarios que va a permitir una póliza hasta la forma de pago que va ser aceptada por los clientes.

Esta aplicación tiene como valor agregado el cálculo de las comisiones devengadas por los promotores, la generación de reportes estadísticos que apoyarán a la gerencia en el proceso de toma de decisiones.

Este sistema de administración de ventas de seguros va a manipular la información de una manera rápida y segura. Hará uso de las herramientas y recursos informáticos de actualidad, por lo que constituirá un avance seguro en el control de la información de la empresa Corredores de Seguros Generales S. A..

1- Alternativa I

1.1- Factibilidad Técnica

Hardware:

- Una computadora (HP PAVILION)
- Procesador Athlon 1.4 Ghz
- 256 Mb de Memoria RAM
- DD 40Gb
- Monitor VGA de 15"
- Teclado y Mouse
- Una impresora HP DeskJet 3420C Printer USB
- Un estabilizador y una batería de 1 hora TRIPP LITE

Software:

- Software de Anitvirus (Norton2003)*
- Sistema Operativo (Window XP)*
- Administrador de Base de Datos incluido en el Microsoft Office 2000 (Microsoft Access)

* Incluido en la compra de la computadora

Recursos Humanos:

Personal para capacitación en el uso del sistema (una persona durante una semana).

Personal para mantenimiento preventivo y correctivo del nuevo sistema (dos personas, la cual estará a las ordenes durante una semana, pasado este periodo el mantenimiento será una política de la empresa).

Personal para el análisis y diseño del sistema (dos personas, durante tres meses).

Factor Cronológico:

Se necesitaran tres meses para el desarrollo del sistema de acuerdo a los datos obtenidos mediante las técnicas de recolección de datos.

Se necesitara una semana adicional para la capacitación del personal en el uso del sistema.

1.2- Factibilidad Económica

Tabla VIII.2- Factibilidad Económica de la Alternativa I. Ver Anexo D

Tipo de Recurso	Recurso	Cantidad	Tiempo Semanas	Precio Total
Hardware	Computadora	1		U\$ 1,639.19
	Impresora	1		U\$ 109.00
	Estabilizador y Bateria	1		U\$ 178.00
Software	MS Office 2000	1		U\$ 200.00
Recursos Humanos	Capacitación	1	1	U\$ 100.00
	Mantenimiento	1	1	U\$ 50.00
	Análisis y Diseño	1	11	U\$ 750.00
IGV 15%				U\$ 453.93
TOTAL				U\$ 3,480.12

1.3- Factibilidad Operativa

El sistema de administración de Ventas de Seguros, por naturaleza es un sistema con una interfaz muy grafica y amigable, lo que le permite que los empleados de la empresa no se resistan a su utilización.

Es muy importante mencionar que las actividades que consumían días para ser elaborada con mas de un empleado, con este nuevo sistema informatico pueden ser elaboradas en minutos por uno solo. El anterior es un ejemplo de lo útil y lo fácil que va ser el uso de el sistema, para los empleados de corredores de Seguros Generales.

2- Alternativa II

2.1- Factibilidad Técnica

Hardware:

- Una computadora (GATE)
- Procesador Athlon 1.8 Ghz
- 256 Mb de Memoria RAM
- DD 40Gb
- Monitor VGA de 15"
- Teclado y Mouse
- Una impresora Canon Bubble-Jet Printer S-200
- Un estabilizador y una batería de 1 hora TRIPP LITE

Software:

- Software de Anitvirus (Norton2003)*
- Sistema Operativo (Window XP)*
- Administrador de Base de Datos incluido en el Microsoft Office 2000 (Microsoft Access).

* Incluido en la compra de la computadora

Recursos Humanos:

Personal para capacitación en el uso del sistema (una persona durante una semana).

Personal para mantenimiento preventivo y correctivo del nuevo sistema (dos personas, la cual estará a las ordenes durante una semana, pasado este periodo el mantenimiento será una política de la empresa).

Personal para el análisis y diseño del sistema (dos personas, durante tres meses).

Factor cronológico:

Se necesitaran tres meses para el desarrollo del sistema de acuerdo a los datos obtenidos mediante las técnicas de recolección de datos.

Se necesitara una semana adicional para la capacitación del personal en el uso del sistema.

2.2- Factibilidad Económica

Tabla VIII.3: Factibilidad Económica de la Alternativa II. Ver Anexo E

Tipo de Recurso	Recurso	Cantidad	Tiempo Semanas	Precio Total
Hardware	Computadora	1		U\$ 1,225.18
	Impresora	1		U\$ 74.00
	Estabilizador y Bateria	1		U\$ 162.94
Software	MS Office 2000	1		U\$ 200.00
Recursos Humanos	Capacitación	1	1	U\$ 100.00
	Mantenimiento	1	1	U\$ 50.00
	Análisis y Diseño	1	11	U\$ 750.00
IGV 15%				U\$ 384.32
TOTAL				U\$ 2,946.44

2.3- Factibilidad Operativa

Como ya se había mencionado los empleados del nuevo sistema se encuentran satisfechos con el mismo, y esta alternativa paralela al estudio de factibilidad planteado anteriormente disminuye los costos y además los empleados y futuros usuarios están motivados con la implementación del nuevo sistema ya sea por la alternativa I o por la alternativa II.

G- Análisis Costo – Beneficio

Una vez teniendo los costos de las dos alternativas procedemos a realizar la evaluación de estas con respecto a los beneficios que generaran.

Es importante mencionar que independientemente de la alternativa elegida el sistema traerá como beneficio un aumento en las ventas entre un 15% y un 20% debido a que permitirá enfocar a uno de los dos empleados que laboran en el área administrativa hacia el área de ventas.

El porcentaje promedio que Corredores de Seguros Generales S.A paga a sus proveedores es del 70% de las ventas netas y del 8% a sus promotores. Es decir que estos varían según la póliza que se esta vendiendo.

Los datos presentados en el estado de resultados comprendido en el periodo de julio/01 a julio/02 son:

Tabla VIII.4: Datos contables de la Empresa

	En Córdobas C\$
Ventas netas	2,354,879.32
Pago a proveedores	(1,648,415.52)
Pago comisiones	(56,517.10)
Pago de salarios	(396,000.00)
Pago de servicios públicos	(18,010.20)
Gastos operativos	(9,906.00)
Utilidad Neta	226,030.50

1- Punto de Equilibrio

El Punto de Equilibrio es el tiempo que tomaría para que el total de ingresos incrementados y/o la reducción de gasto se igual al costo total. Sin embargo, no toma en cuenta el valor del dinero en el tiempo.

$$PE = (\text{Costo} / \text{Ingresos Incrementados y/o Reducción de Gastos}) \times 12 \text{ (meses)}$$

Tabla VIII.5: Datos proyectados al termino de un año

Alternativa I	
Rubro	Valor en córdobas
Valor del sistema (Tasa 14.80)	51,505.77
Ventas incrementadas en 15%	2,708,111.21
Pago a proveedores	(1,895,677.84)
Utilidad antes de gastos	812,433.37
Pago comisiones	(64,994.66)
Gastos	(423,916.20)

Utilidad Neta	323,522.51
Incremento de la utilidad neta anual	97,492.01

$$PE = (C\$ 51,505.77 / C\$ 97,492.01) \times 12 \text{ meses}$$

$$PE = 6.34 \text{ meses}$$

Tabla VIII.6: Datos proyectados al termino de un año

Alternativa II	
Rubro	Valor en córdobas
Valor del sistema (Tasa 14.80)	43,607.31
Ventas incrementadas en 15%	2,708,111.21
Pago a proveedores	(1,895,677.84)
Utilidad antes de gastos	812,433.37
Pago comisiones	(64,994.66)
Gastos	(423,916.20)
Utilidad Neta	323,522.51
Incremento de la utilidad neta anual	97,492.01

$$PE = (C\$ 43,607.31 / C\$ 97,492.01) \times 12 \text{ meses}$$

$$PE = 5.37 \text{ meses}$$

2- Análisis Comparativo de las Alternativas

Como se puede observar en este estudio de factibilidad, cualquiera de las dos alternativas presentadas anteriormente resolverían el problema que actualmente atraviesa Corredores de Seguros Generales S. A., pero el objetivo de este trabajo es presentar a nuestro cliente la alternativa más viable para cumplir con los objetivos propuestos en este proyecto.

La alternativa I, nos da la oportunidad de adquirir un equipo de mejor calidad y la alternativo II indica ajustar nuestros requerimientos de Hardware con un equipo de menor calidad. Como se trata de un sistema que se pretende que en un futuro sea

compartido por varios usuarios (en Red), lo mas recomendable es adquirir el equipo de la alternativa I el cual después será ocupado como servidor.

La alternativa I cuesta U\$ 533.68 mas que la alternativa II debido que el equipo de la Alternativa I es fabricado por una empresa de mayor prestigio y la computadora de la Alternativa II es clon.

3- Dictamen

Con el análisis obtenido entre las dos alternativas presentadas en este proyecto se ha dictaminado que la opción más viable para Corredores Seguros Generales S. A. es la alternativa I la cual se ajusta mas al desarrollo de la Empresa aunque se incurra en aproximadamente un mes más que en la alternativa II para recuperar la inversión.

H- Ambiente de Trabajo

Modelo del Procesador: El Sistema de Administración de Ventas de Seguros será utilizado por un sistema monoprocesador, debido a que este no exige grandes capacidades de computo.

Modelo del Usuario: Este sistema será monousuario debido a que aún no existe la necesidad de compartir datos en los diferentes departamentos. Si esto llegará a ocurrir el sistema está preparado para trabajar en un entorno multiusuario.

Medio ambiente: En base a la inspección realizada en la oficina de computo, lugar donde estará la computadora que ejecutará el sistema, se llegó a la conclusión de que se presta apropiadamente el medio ambiente que se necesita para explotar al máximo las características de esta aplicación sin que se vea afectada por el rendimiento de la computadora en la que se ejecutará.

V- ANÁLISIS ORIENTADO A PROCESOS DEL SISTEMA DE ADMINISTRACIÓN DE VENTAS DE SEGUROS EN LA EMPRESA CORREDORES DE SEGUROS GENERALES S.A

A- Diagramas de Flujo de Datos (DFD)

1- Diagrama de Contexto

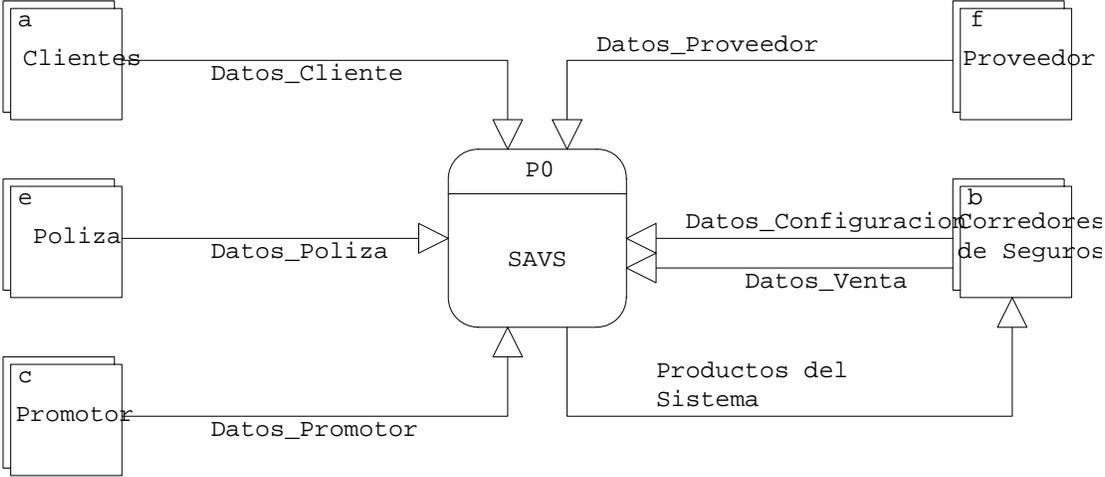


Figura IX.1: Diagrama de Contexto (Nivel 0)

2- Nivel 1

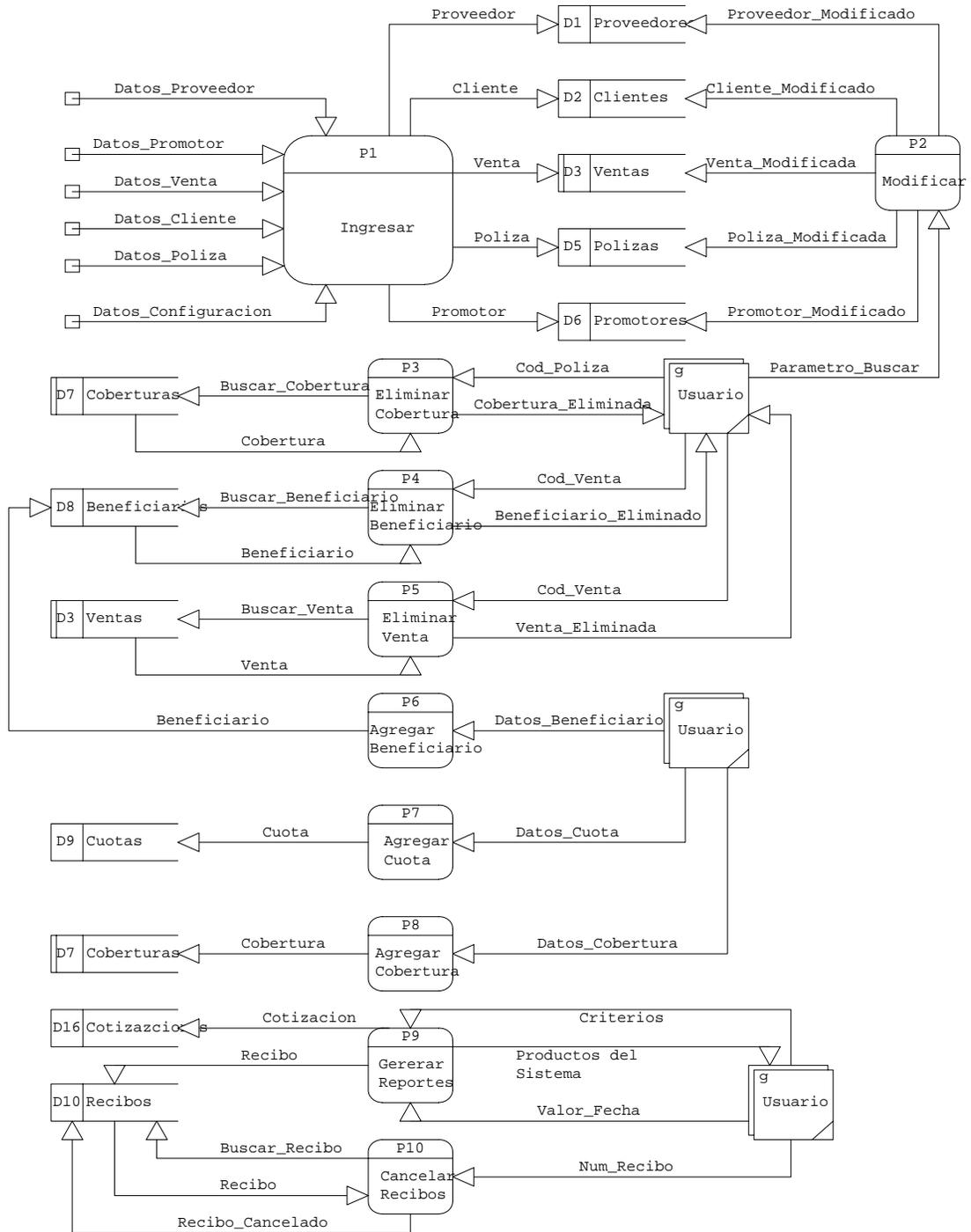


Figura IX.2: Diagrama Nivel 1

3- Nivel 2

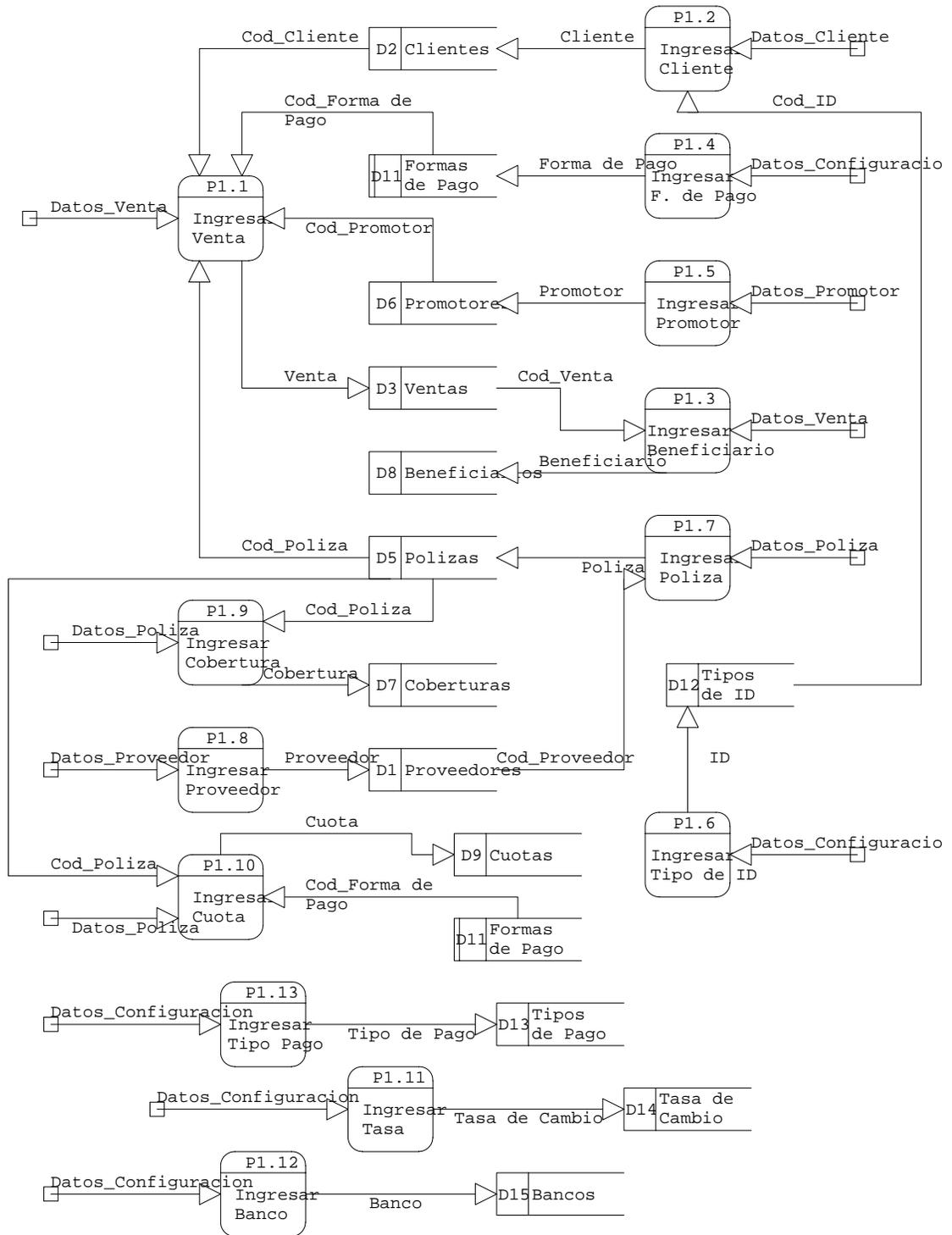


Figura IX.3: Diagrama Nivel 2

4- Nivel 3

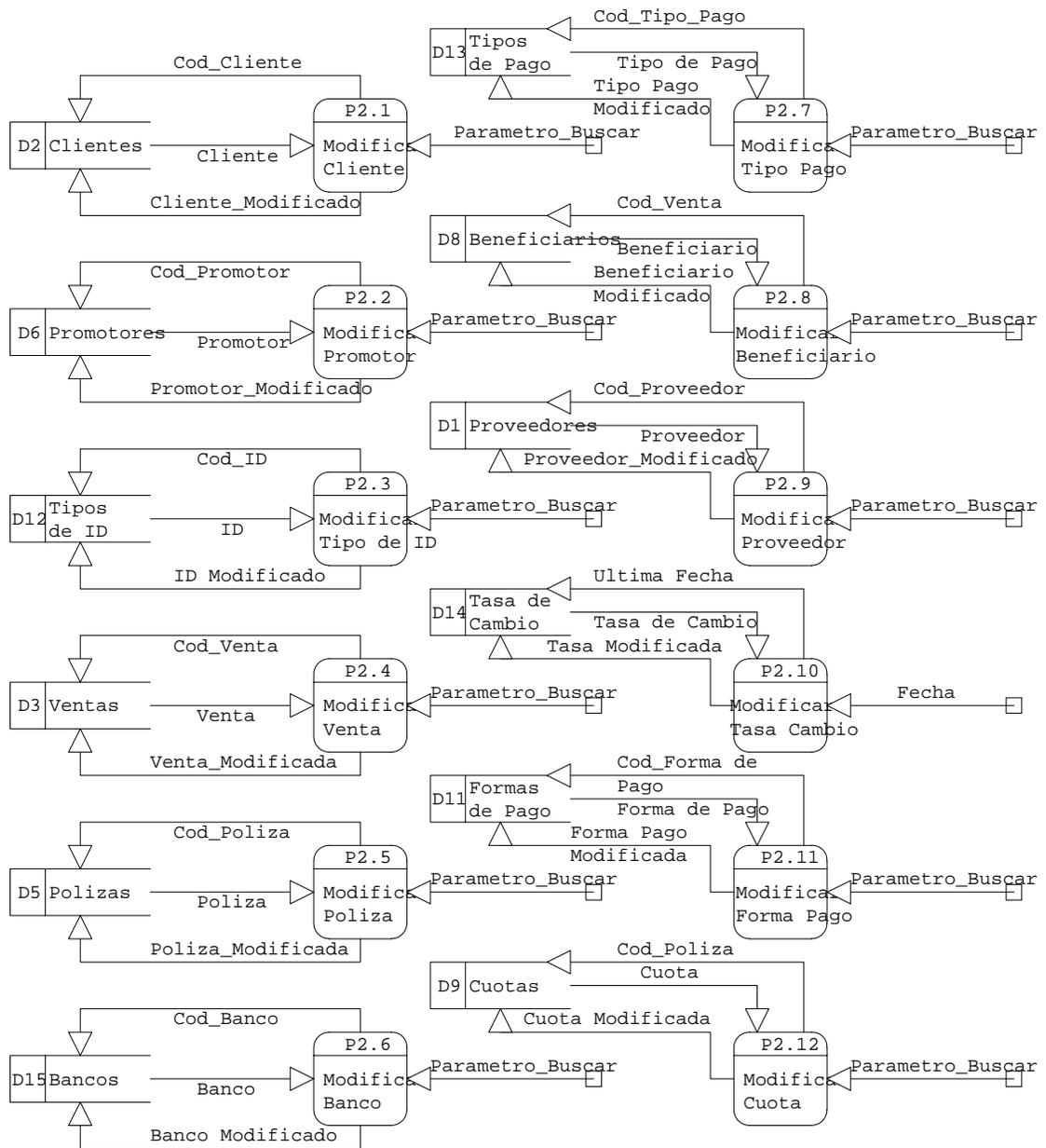


Figura IX.4: Diagrama Nivel 3

B- Especificaciones de Procesos

1- Nivel 1

Eliminar Cobertura

Ingresar código de la póliza

SI EXISTE

Mostrar coberturas disponibles

Elegir cobertura a eliminar

Eliminar cobertura

SINO

Enviar mensaje “Esta póliza no existe”

Eliminar Beneficiario

Ingresar código de la venta

SI EXISTE

Mostrar los beneficiarios

Elegir beneficiario a eliminar

Eliminar beneficiario

SINO

Enviar mensaje “Esta venta no existe”

Eliminar Venta

Ingresar código de la venta

SI EXISTE

Mostrar datos de la venta

SI FECHA DE LA VENTA <> FECHA DE HOY

Eliminar venta

SINO

Enviar mensaje “Esta venta no puede ser eliminada”

SINO

Enviar mensaje “Esta venta no existe”

Agregar Beneficiario

Ingresar código de la venta al que pertenece el beneficiario

SI EXISTE Y LA CANTIDAD DE BENEFICIARIOS ACTUAL <
CANTIDAD DE BENEFICIARIOS PERMITIDOS POR LA POLIZA

Ingresar beneficiario

SI DATOS COMPLETOS Y BENEFICIARIO NO REPETIDO

Guardar beneficiario

SINO

Enviar mensaje "Beneficiario no guardado"

SINO

Enviar mensaje "Esta venta no existe o es el límite de beneficiarios"

Agregar Cuota

Ingresar código de la póliza al que desea agregar una cuota

SI EXISTE

Ingresar cuota

SI LOS DATOS NO SON REPETIDOS

Guardar cuota

SINO

Enviar mensaje "Cuota no guardada"

SINO

Enviar mensaje "Esta póliza no existe"

Agregar Cobertura

Ingresar código de la póliza al que desea agregar una cobertura

SI EXISTE

Ingresar cobertura

SI LOS DATOS NO SON REPETIDOS

Guardar cobertura

SINO

Enviar mensaje "Cobertura no guardada"

SINO

Enviar mensaje "Esta póliza no existe"

Cancelar Recibo

Ingresar el número del recibo

SI EXISTE Y NO HA SIDO CANCELADO

Mostrar los datos del recibo

Cancelar recibo

SI DATOS DE LA CANCELACIÓN ESTAN COMPLETOS Y CORRECTOS

Guardar recibo

SINO

Enviar mensaje “Datos inválidos”

SINO

Enviar mensaje “El recibo no existe o está cancelado”

Generar Reportes

Ingresar los criterios por los cuales desea los reportes

Se muestran los diferentes tipos de reportes en dependencia de los criterios

Se eligen los criterios deseados

SI LOS CRITERIOS SON VALIDOS

Generar reportes

SINO

Enviar mensaje “Criterios inválidos

La generación de recibos recibe el criterio automáticamente. El sistema los crea y los guarda sin mostrarlos al usuario

2- Nivel 2

Ingresar Venta

Ingresar datos de la venta necesarios

SI LOS DATOS ESTAN COMPLETOS

Guardar venta

Llamar al proceso **Ingresar Beneficiarios**

SI NO

Enviar mensaje "Venta no Guardada"

Ingresar Beneficiario

SI LA CANTIDAD DE BENEFICIARIO DE LA VENTA < CANTIDAD
BENEFICIARIO ACEPTADA POR LA PÓLIZA

Ingresar datos del beneficiario

SI LOS DATOS ESTAN COMPLETOS Y NO DUPLICADOS

Guardar beneficiarios

Llamar Generar Reporte

Llamar Cancelar Recibo

SI NO

Enviar mensaje "Beneficiario no Guardado"

SI NO

Enviar mensaje "No se permiten mas beneficiarios para esta venta"

Ingresar Cliente

Ingresar los datos del Cliente

SI LOS DATOS ESTAN COMPLETO Y EL NOMBRE NO SE REPITE

Guardar cliente

SI NO

Enviar mensaje "Cliente no Guardado"

Ingresar Tipo de ID

Ingresar los datos del Tipo de ID

SI LOS DATOS ESTAN COMPLETO Y LA ID NO SE REPITE

Guardar Tipo de ID

SI NO

Enviar mensaje “Tipo de ID no Guardado”

Ingresar Forma de Pago

Ingresar los datos de la Forma de Pago

SI LOS DATOS ESTAN COMPLETO Y LA FORMA DE PAGO NO SE REPITE

Guardar Forma de Pago

SI NO

Enviar mensaje “Tipo de Forma de Pago no Guardada”

Ingresar Tipo Pago

Ingresar los datos del Tipo de Pago

SI LOS DATOS ESTAN COMPLETO Y EL TIPO DE PAGO NO SE REPITE

Guardar Tipo de Pago

SI NO

Enviar mensaje “Tipo de Pago no Guardado”

Ingresar Banco

Ingresar los datos de Banco

SI LOS DATOS ESTAN COMPLETO Y EL BANCO NO SE REPITE

Guardar Banco

SI NO

Enviar mensaje “Banco no Guardado”

Ingresar Tasa de Cambio

Ingresar los datos de Tasa de Cambio

Guardar Tasa de Cambio

Ingresar Proveedor

Ingresar los datos del Proveedor

SI LOS DATOS ESTAN COMPLETO Y EL NOMBRE NO SE REPITE

Guardar Proveedor

SI NO

Enviar mensaje “Proveedor no Guardado”

Ingresar Promotor

Ingresar los datos del Promotor

SI LOS DATOS ESTAN COMPLETO Y EL NOMBRE NO SE REPITE

Guardar Promotor

SI NO

Enviar mensaje “Promotor no Guardado”

Ingresar Póliza

Ingresar los datos de la Póliza

SI LOS DATOS ESTAN COMPLETO Y LA POLIZA NO SE REPITE

Guardar Póliza

Llamar Ingresar Cobertura**Llamar Ingresar Cuota**

SI NO

Enviar mensaje “Póliza no Guardada”

Ingresar Cobertura

Ingresar los datos de la Cobertura

SI LOS DATOS ESTAN COMPLETO Y LA COBERTURA NO SE REPITE

Guardar Cobertura

SI NO

Enviar mensaje “Cobertura no Guardada”

Ingresar Cuota

Ingresar los datos de la Cuota

SI LOS DATOS ESTAN COMPLETO Y LA CUOTA NO SE REPITE

Guardar CUOTA

SI NO

Enviar mensaje “Cuota no Guardada”

3- Nivel 3

Modificar Cliente

Introducir el Código del Cliente

SI EXISTE

Mostrar los datos del Cliente

Modificar Cliente

SI LOS DATOS ESTAN COMPLETOS Y EL NOMBRE NO SE REPITE

Guardar los cambios

SI NO

Eliminar los cambios

SI NO

Enviar mensaje "Cliente no Existe"

Modificar Promotor

Introducir el Código del Promotor

SI EXISTE

Mostrar los datos del Promotor

Modificar Promotor

SI LOS DATOS ESTAN COMPLETOS Y EL NOMBRE NO SE REPITE

Guardar los cambios

SI NO

Eliminar los cambios

SI NO

Enviar mensaje "Promotor no Existe"

Modificar Proveedor

Introducir el Código del Proveedor

SI EXISTE

Mostrar los datos del Proveedor

Modificar Proveedor

SI LOS DATOS ESTAN COMPLETOS Y EL NOMBRE NO SE REPITE

Guardar los cambios

SI NO

Eliminar los cambios

SI NO

Enviar mensaje “Proveedor no Existe”

Modificar Póliza

Introducir el Código de la Póliza

SI EXISTE

Mostrar los datos de la Póliza

Modificar Póliza

SI LOS DATOS ESTAN COMPLETOS Y LA PÓLIZA NO SE REPITE

Guardar los cambios

SI NO

Eliminar los cambios

SI NO

Enviar mensaje “Póliza no Existe”

Modificar Banco

Introducir el Código del Banco

SI EXISTE

Mostrar los datos del Banco

Modificar Banco

SI LOS DATOS ESTAN COMPLETOS Y EL BANCO NO SE REPITE

Guardar los cambios

SI NO

Eliminar los cambios

SI NO

Enviar mensaje “Banco no Existe”

Modificar Tipo de ID

Introducir el Código del Tipo de ID

SI EXISTE

Mostrar los datos del Tipo de ID

Modificar Tipo de ID

SI LOS DATOS ESTAN COMPLETOS Y EL TIPO DE ID NO SE REPITE

Guardar los cambios

SI NO

Eliminar los cambios

SI NO

Enviar mensaje “Tipo de ID no Existe”

Modificar Tipo Pago

Introducir el Código del Tipo de Pago

SI EXISTE

Mostrar los datos del Tipo de Pago

Modificar Tipo de Pago

SI LOS DATOS ESTAN COMPLETOS Y EL TIPO DE PAGO NO SE REPITE

Guardar los cambios

SI NO

Eliminar los cambios

SI NO

Enviar mensaje “Tipo de Pago no Existe”

Modificar Forma de Pago

Introducir el Código de la Forma de Pago

SI EXISTE

Mostrar los datos de la Forma de Pago

Modificar Forma de Pago

SI LOS DATOS ESTAN COMPLETOS Y LA FORMA DE PAGO NO SE REPITE

Guardar los cambios

SI NO

Eliminar los cambios

SI NO

Enviar mensaje “Forma de pago no Existe”

Modificar Venta

Introducir el Código de la Venta

SI EXISTE

Mostrar los datos de la Venta

Modificar Venta

SI LOS DATOS ESTAN COMPLETOS

Guardar los cambios

SI NO

Eliminar los cambios

SI NO

Enviar mensaje "Venta no Existe"

Modificar Tasa de Cambio

Recuperar ultima actualización

Modificar tasa

Guardar tasa

Modificar Beneficiario

Introducir el Código de la Venta

SI EXISTE

Mostrar los Beneficiarios de esa Venta

Modificar Beneficiarios

SI LOS DATOS ESTAN COMPLETOS Y EL NOMBRE NO SE REPITE

Guardar los cambios

SI NO

Eliminar los cambios

SI NO

Enviar mensaje "Venta no Existe"

Modificar Cobertura

Introducir el Código de la Póliza

SI EXISTE

Mostrar las Cobertura de esa Póliza

Modificar Cobertura

SI LOS DATOS ESTAN COMPLETOS Y LA COBERTURA NO SE REPITE

Guardar los cambios

SI NO

Eliminar los cambios

SI NO

Enviar mensaje "Póliza no Existe"

C- Diccionario de Datos

1- Entidades Externas

Nombre: Clientes

Numeral: a

Descripción: Son todos los clientes de Corredores de Seguros Generales S. A.

Nombre: Corredores de Seguros

Numeral: b

Descripción: Son los directores de Corredores de Seguros Generales S. A.

Nombre: Promotor

Numeral: c

Descripción: Son todos los promotores de Corredores de Seguros Generales S. A.

Nombre: Póliza

Numeral: e

Descripción: Son todas las diferentes pólizas que oferta Corredores de Seguros Generales S. A.

Nombre: Proveedor

Numeral: f

Descripción: Son todos los proveedores de póliza para Corredores de Seguros Generales S. A.

Nombre: Usuario

Numeral: g

Descripción: Es el usuario del Sistema de Administración de Ventas de Seguros

2- Procesos

Nombre: SAVS

Numeral: P0

Descripción: Es el proceso representativo de lo que hace básicamente el sistema

Nombre: Ingresar

Numeral: P1

Descripción: Proceso en el que se ingresan todos los datos al sistema

Nombre: Modificar

Numeral: P2

Descripción: Proceso en el que se modifican todos los datos del sistema

Nombre: Eliminar Cobertura

Numeral: P3

Descripción: Proceso en el que se elimina determinada cobertura de una póliza

Nombre: Eliminar Beneficiario

Numeral: P4

Descripción: Proceso en el que se elimina determinado beneficiario de una venta

Nombre: Eliminar Venta

Numeral: P5

Descripción: Proceso en el que se elimina una venta determinada

Nombre: Agregar Beneficiario

Numeral: P6

Descripción: Proceso en el que se agrega un beneficiario a determinada venta

Nombre: Agregar cuota

Numeral: P7

Descripción: Proceso en el que se agrega una cuota a determinada póliza

Nombre: Agregar cobertura

Numeral: P8

Descripción: Proceso en el que se agrega una cobertura a determinada póliza

Nombre: Generar Reporte

Numeral: P9

Descripción: Proceso en el que se generan los diferentes reportes al usuario del sistema

Nombre: Cancelar Recibo

Numeral: P10

Descripción: Proceso en el que se cancelan los recibos de las ventas efectuadas

Nombre: Ingresar Venta

Numeral: P1.1

Descripción: Proceso en el que se ingresan las ventas al sistema

Nombre: Ingresar Cliente

Numeral: P1.2

Descripción: Proceso en el que se ingresan los clientes al sistema

Nombre: Ingresar Beneficiario

Numeral: P1.3

Descripción: Proceso en el que se ingresan los beneficiarios de las ventas al sistema

Nombre: Ingresar F. de Pago

Numeral: P1.4

Descripción: Proceso en el que se ingresan las formas de pago al sistema

Nombre: Ingresar Promotor

Numeral: P1.5

Descripción: Proceso en el que se ingresan los promotores al sistema

Nombre: Ingresar Tipo de ID

Numeral: P1.6

Descripción: Proceso en el que se ingresan las diferentes identificaciones al sistema

Nombre: Ingresar Póliza

Numeral: P1.7

Descripción: Proceso en el que se ingresan las pólizas al sistema

Nombre: Ingresar Proveedor

Numeral: P1.8

Descripción: Proceso en el que se ingresan los proveedores al sistema

Nombre: Ingresar Cobertura

Numeral: P1.9

Descripción: Proceso en el que se ingresan las coberturas de una póliza al sistema

Nombre: Ingresar Cuota

Numeral: P1.10

Descripción: Proceso en el que se ingresan las cuotas de una póliza al sistema

Nombre: Ingresar Tasa

Numeral: P1.11

Descripción: Proceso en el que se ingresa la tasa de cambio del día al sistema

Nombre: Ingresar Banco

Numeral: P1.12

Descripción: Proceso en el que se ingresan los bancos al sistema

Nombre: Ingresar Tipo Pago

Numeral: P1.13

Descripción: Proceso en el que se ingresan los tipos de pago al sistema

Nombre: Modificar Cliente

Numeral: P2.1

Descripción: Proceso en el que se modifican los datos del cliente

Nombre: Modificar Promotor

Numeral: P2.2

Descripción: Proceso en el que se modifican los datos del promotor

Nombre: Modificar Tipo de ID

Numeral: P2.3

Descripción: Proceso en el que se modifican los tipos de ID

Nombre: Modificar Venta

Numeral: P2.4

Descripción: Proceso en el que se modifican los datos de la venta

Nombre: Modificar Póliza

Numeral: P2.5

Descripción: Proceso en el que se modifican los datos de la póliza

Nombre: Modificar Banco

Numeral: P2.6

Descripción: Proceso en el que se modifican los datos del banco

Nombre: Modificar Tipo Pago

Numeral: P2.7

Descripción: Proceso en el que se modifican los datos del tipo de pago

Nombre: Modificar Beneficiario

Numeral: P2.8

Descripción: Proceso en el que se modifican los datos del beneficiario de una venta

Nombre: Modificar Proveedor

Numeral: P2.9

Descripción: Proceso en el que se modifican los datos del proveedor

Nombre: Modificar Tasa Cambio

Numeral: P2.10

Descripción: Proceso en el que se modifican la tasa de cambio del día

Nombre: Modificar Forma Pago

Numeral: P2.11

Descripción: Proceso en el que se modifican los datos de la forma de pago

Nombre: Modificar Cuota

Numeral: P2.12

Descripción: Proceso en el que se modifican los datos de la cuota de una póliza

3- Almacenes de Datos

Nombre: Proveedores

Numeral: D1

Descripción: Almacena los datos de los proveedores de pólizas

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	ID_PROVEEDOR	Auto numérico		Código del proveedor
2	PROVEEDOR	Texto	50	Nombre del proveedor
3	DIRECCIÓN	Texto	100	Dirección del proveedor
4	CONTACTO	Texto	100	Nombre del contacto
5	TELEFONO	Texto	10	# de Teléfono del proveedor
6	EXTENSIÓN	Entero	5	Extensión del # de Tel.

Nombre: Clientes

Numeral: D2

Descripción: Almacena los datos de los clientes

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	ID_CLIENTE	Auto numérico		Código del cliente
2	FECHA_VISITA	Fecha		Fecha de la visita
3	NOMBRE	Texto	70	Nombre del Cliente
4	DIRECCIÓN	Texto	100	Dirección del cliente
5	TELEFONO	Texto	10	# de Teléfono del cliente
6	CELULAR	Texto	10	# de Celular del cliente
7	TIPO_ID	Entero	3	Código del tipo de identificación
8	NUM_ID	Texto	20	Número de la identificación

Nombre: Ventas

Numeral: D3

Descripción: Almacena los datos de las ventas de pólizas

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	ID_VENTA	Auto numérico		Código de la venta
2	ID_CLIENTE	Entero	4	Código del cliente
3	ID_POL	Texto	20	Código de la póliza
4	ID_PROMOTOR	Entero	3	Código del promotor
5	FECHA_ADQUI	Fecha		Fecha de adquisición
6	FORMA_PAGO	Entero	2	Forma de pago
7	ACEPTADO	Sí/No		Sí la venta esta activa

Nombre: Pólizas

Numeral: D5

Descripción: Almacena los datos de las pólizas

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	ID_PROVEEDOR	Entero	2	Código del proveedor
2	ID_POL	Texto	20	Código de la póliza
3	DESCRIPCION	Texto	150	Nombre de la póliza
4	VALOR_COSTO	Real	10	Valor del costo
5	CANT_BENEFIC	Entero	2	Cantidad de beneficiarios
6	VALOR_VENTA	Real	10	Valor de venta
7	PORC_COMISION	Real	3	Porcentaje de comisión

Nombre: Promotores

Numeral: D6

Descripción: Almacena los datos de los promotores

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	ID_PROMOTOR	Auto numérico		Código del promotor
2	NOMBRE	Texto	70	Nombre
3	DIRECCION	Texto	100	Dirección
4	TELEFONO	Texto	10	Teléfono
5	CELULAR	Texto	10	Celular
6	FECHA_INGRESO	Fecha		Fecha de ingreso
7	SALARIO	Real	10	Salario del promotor

Nombre: Coberturas

Numeral: D7

Descripción: Almacena los datos de las coberturas de las pólizas

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	ID_POL	Texto	20	Código de la póliza
2	COBERTURA	Texto	50	Cobertura
3	VALOR_COBER	Real	10	Valor a pagar de la cobertura

Nombre: Beneficiarios

Numeral: D8

Descripción: Almacena los datos de los beneficiarios de las ventas

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	ID_VENTA	Auto numérico		Código de la venta
2	BENEFICIARIO	Texto	70	Nombre del beneficiario

Nombre: Cuotas

Numeral: D9

Descripción: Almacena los datos de las cuotas de las pólizas

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	ID_POL	Texto	50	Código de la póliza
2	FORMA_PAGO	Entero	2	Forma de pago
3	VALOR_CUOTA	Real	10	Valor de la cuota

Nombre: Recibos

Numeral: D10

Descripción: Almacena los datos de los recibos

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	ID_RECIBO	Auto numérico		Número del recibo
2	ID_VENTA	Entero	3	Código de la venta
3	VALOR_RECIBO	Real	10	Valor del recibo
4	VALOR_LETRA	Texto	100	Valor en letras
5	PORC_COMISION	Real	3	Porcentaje de comisión
6	DEL	Fecha		Fecha inicial del periodo
7	AL	Fecha		Fecha final del periodo
8	CUOTA_NUM	Entero	2	Número de la cuota
9	FECHA_CANCEL	Fecha		Fecha de cancelación
10	TIPO_PAGO	Entero	2	Tipo de pago
11	ID_BANCO	Entero	2	Código del banco
12	REFERENCIA	Entero	7	# de Referencia
13	ESTADO	Texto	10	Estado del recibo
14	FECHA_EMISIÓN	Fecha		Fecha de emisión
15	DÓLAR	Real	10	Pago en dólares
16	CORDOBA	Real	10	Pago en córdobas
17	TASA	Real	4	Tasa de cambio
18	VALOR_RECIBO_CORDOBA	Real	10	Valor en córdobas

Nombre: Formas de Pago

Numeral: D11

Descripción: Almacena las formas de pago disponibles

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	FORMA_PAGO	Auto numérico		Código de la forma de pago
2	DESCRIPCIÓN	Texto	50	Nombre de la forma de pago
3	CANT_PAGOS	Entero	2	Cantidad de pagos

Nombre: Tipos de ID

Numeral: D12

Descripción: Almacena los datos de los tipos de identificaciones

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	TIPO_ID	Auto numérico		Código del tipo de identificación
2	DESCRIPCIÓN	Texto	25	Nombre de la identificación

Nombre: Tipos de Pago

Numeral: D13

Descripción: Almacena los datos de los diferentes tipos de pago aceptados

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	ID TIPO PAGO	Auto numérico		Código del tipo de pago
2	TIPO_PAGO	Texto	20	Nombre del tipo de pago

Nombre: Tasa de Cambio

Numeral: D14

Descripción: Almacena los datos de las tasas de cambio

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	FECHA	Fecha		Fecha de la tasa de cambio
2	TASA	Real	4	Tasa de cambio

Nombre: Bancos

Numeral: D15

Descripción: Almacena los datos de los bancos

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	ID BANCO	Auto numérico		Código del banco
2	BANCO	Texto	50	Nombre del banco

Nombre: Cotizaciones

Numeral: D16

Descripción: Almacena los datos de las cotizaciones elaboradas

Contenido:

N° del Campo	Nombre del Campo	Tipo	Tamaño	Descripción
1	ID_COTIZACIÓN	Auto numérico		Código de la cotización
2	ID_CLIENTE	Entero	3	Código del cliente
3	ID_POL	Texto	50	Código de la póliza
4	FECHA	Fecha		Fecha de la cotización
5	CONTACTO	Texto	50	Nombre del contacto
6	ID_POMOTOR	Entero	2	Código del promotor

4- Flujos de Datos

Nombre: Banco

Descripción: Contiene el registro banco

Nombre: Banco_Modificado

Descripción: Contiene el registro banco cuando ha sido modificado

Nombre: Beneficiario

Descripción: Contiene el registro beneficiario

Nombre: Beneficiario Modificado

Descripción: Contiene el registro beneficiario cuando ha sido modificado

Nombre: Beneficiario_Eliminado

Descripción: Es un mensaje de notificación al usuario

Nombre: Buscar_Beneficiario

Descripción: Contiene el código de la venta al cual pertenece el beneficiario

Nombre: Buscar_Cobertura

Descripción: Contiene el código de la póliza al cual pertenece la cobertura

Nombre: Buscar_Recibo

Descripción: Contiene el código del recibo a cancelar

Nombre: Buscar_Venta

Descripción: Contiene el código de la venta

Nombre: Cliente

Descripción: Contiene el registro cliente

Nombre: Cliente_Modificado

Descripción: Contiene el registro cliente cuando ha sido modificado

Nombre: Cobertura

Descripción: Contiene el registro cobertura

Nombre: Cobertura_Eliminada

Descripción: Es un mensaje de notificación al usuario

Nombre: Cod_Banco

Descripción: Contiene el código del banco

Nombre: Cod_Cliente

Descripción: Contiene el código del cliente

Nombre: Cod_Forma de Pago

Descripción: Contiene el código de la forma de pago

Nombre: Cod_ID

Descripción: Contiene el código de la identificación

Nombre: Cod_Poliza

Descripción: Contiene el código de la póliza

Nombre: Cod_Promotor

Descripción: Contiene el código del promotor

Nombre: Cod_Proveedor

Descripción: Contiene el código del proveedor

Nombre: Cod_Tasa_Cambio

Descripción: Contiene la fecha de la última tasa de cambio

Nombre: Cod_Tipo_Pago

Descripción: Contiene el código del tipo de pago

Nombre: Cod_Venta

Descripción: Contiene el código de la venta

Nombre: Cotización

Descripción: Contiene los datos de la cotización

Nombre: Criterios

Descripción: Contiene los diferentes parámetros para la generación de reportes

Nombre: Cuota

Descripción: Contiene el registro cuota

Nombre: Cuota Modificada

Descripción: Contiene e registro cuota cuando ha sido modificada

Nombre: Datos_Beneficiario

Descripción: Contiene los datos de un beneficiarios

Nombre: Datos_Cliente

Descripción: Contiene los datos de un cliente

Nombre: Datos_Cobertura

Descripción: Contiene los datos de una cobertura

Nombre: Datos_Configuracion

Descripción: Contiene los datos de la configuración del sistema

Nombre: Datos_Cuota

Descripción: Contiene los datos de una cuota

Nombre: Datos_Poliza

Descripción: Contiene los datos de una póliza

Nombre: Datos_Promotor

Descripción: Contiene los datos de un promotor

Nombre: Datos_Venta

Descripción: Contiene los datos de una venta

Nombre: Fecha

Descripción: Contiene la fecha actual del sistema

Nombre: Forma de Pago

Descripción: Contiene el registro forma de pago

Nombre: Forma Pago Modificada

Descripción: Contiene el registro Forma de Pago cuando ha sido modificado

Nombre: ID

Descripción: Contiene el registro identificación

Nombre: ID_Modificado

Descripción: Contiene el registro ID cuando ha sido modificado

Nombre: Num_Recibo

Descripción: Contiene el número del recibo

Nombre: Parámetro_Buscar

Descripción: Contiene un parámetro de búsqueda

Nombre: Póliza

Descripción: Contiene el registro póliza

Nombre: Póliza_Modificada

Descripción: Contiene el registro póliza cuando ha sido modificado

Nombre: Productos del Sistema

Descripción: Contiene los diferentes tipos de reportes que brinda el sistema

Nombre: Promotor

Descripción: Contiene el registro promotor

Nombre: Promotor_Modificado

Descripción: Contiene el registro promotor cuando ha sido modificado

Nombre: Proveedor

Descripción: Contiene el registro proveedor

Nombre: Proveedor_Modificado

Descripción: Contiene el registro proveedor cuando ha sido modificado

Nombre: Recibo

Descripción: Contiene el registro recibo

Nombre: Recibo_Cancelado

Descripción: Contiene el registro recibo una vez que ha sido cancelado

Nombre: Tasa de Cambio

Descripción: Contiene el registro tasa de cambio del día

Nombre: Tasa Modificada

Descripción: Contiene el registro tasa de cambio del día cuando ha sido modificado

Nombre: Tipo de Pago

Descripción: Contiene el registro tipo de pago

Nombre: Tipo Pago_Modificado

Descripción: Contiene el registro tipo de pago cuando ha sido modificado

Nombre: Ultima Fecha

Descripción: Contiene la fecha de la última tasa de cambio ingresada

Nombre: Valor_Fecha

Descripción: Contiene la fecha actual del sistema

Nombre: Venta

Descripción: Contiene el registro venta

Nombre: Venta_Eliminada

Descripción: Es un mensaje de notificación al usuario

Nombre: Venta_Modificada

Descripción: Contiene el registro venta cuando ha sido modificado

VI- DISEÑO ORIENTADO A PROCESOS DEL SISTEMA DE ADMINISTRACIÓN DE VENTAS DE SEGUROS EN LA EMPRESA CORREDORES DE SEGUROS GENERALES S.A

A- Descripción del Sistema

El Sistema de Administración de Ventas de Seguros estará compuesto por una barra de menú principal, el cual constará de las siguientes opciones:

SAVS Datos Póliza Recibo Cotización Configurar Misceláneos Reportes Imprimir

Cada opción contiene menús desplegados, los cuales se describen a continuación:

Tabla X.1: Barra de Menú Principal del Sistema

SAVS	Respaldar	
	Salir	
Datos	Cliente	Ingresar
		Modificar
		Visualizar
	Promotores	Ingresar
		Modificar
	Proveedores	Ingresar
Modificar		
Póliza	Comprar	Ingresar
		Modificar
	Vender	Ingresar
		Modificar
		Visualizar
Recibo	Cancelar	
	Revertir	
	Imprimir	
Cotización	Generar	
	Imprimir	

Configurar	Empresa	
	Banco	Ingresar
		Modificar
	Forma de Pago	Ingresar
		Modificar
	Modalidad de Pago	Ingresar
		Modificar
	Tasa de Cambio	Modificar
Tipo de Identificación	Ingresar	
	Modificar	
Misceláneos	Beneficiario	Ingresar
		Modificar
	Cobertura	Ingresar
		Modificar
	Cuota	Ingresar
Reportes	Comisiones	
	Estado de Cuenta	
	Listado de Clientes	
	Reportes de Ventas	
Imprimir		

B-Diagrama Entidad / Relación

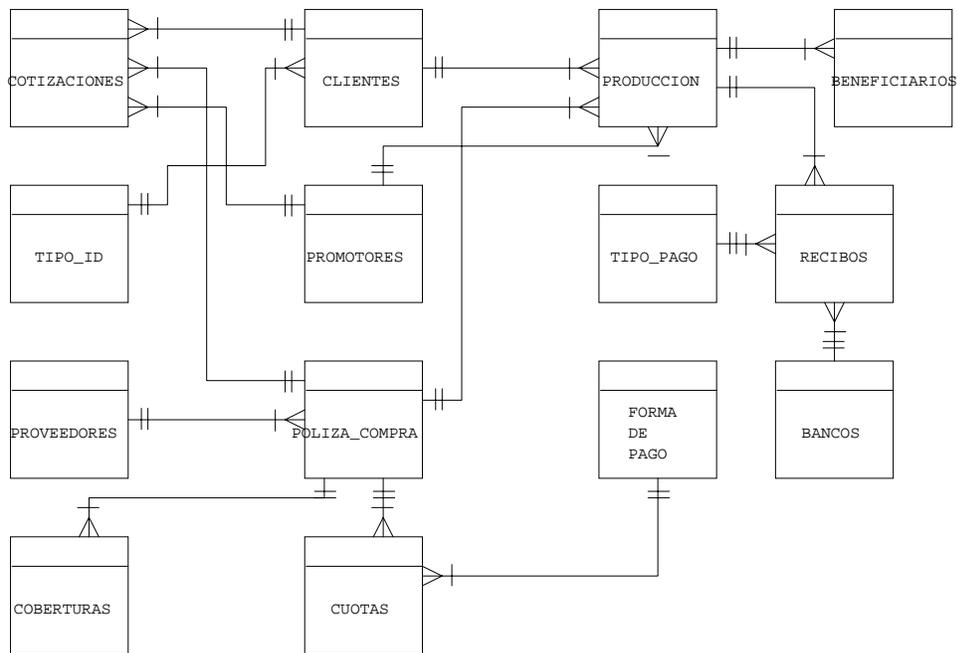


Figura X.1: Diagrama Entidad / Relación del Sistema

C- Formatos de Entrada

Las entradas al sistema serán a través de formularios, los cuales se describen a continuación:

Formulario Ingresar Banco

Código	<input type="text" value="Autonómico"/>
Banco	<input type="text"/>
 	

Formulario Ingresar Beneficiario

N° de la Venta	<input type="text"/>
Código de la Póliza	<input type="text"/>
Beneficiario	<input type="text"/>
Cantidad de Beneficiarios	<input type="text"/>
 	

Formulario Cancelar Recibo

N° de Recibo	<input type="text"/>	Nombre	<input type="text"/>
Total en Dolares	<input type="text"/>	Modalidad de Pago	<input type="text"/>
Total en Cordobas	<input type="text"/>	Banco	<input type="text"/>
		Referencia	<input type="text"/>
$\text{Total Recibo} = (\text{Dolares} \times \text{Tasa}) + \text{Cordobas}$			
	<input type="text"/>	<input type="text"/>	<input type="text"/>
Valor en Letras	<input type="text"/>		
 			

Formulario Ingresar Cliente

Código del Cliente	<input type="text" value="9"/>
Fecha de Visita	<input type="text"/>
Nombre	<input type="text"/>
Dirección	<input type="text"/>
Teléfono	<input type="text"/>
Celular	<input type="text"/>
Identificación	<input type="text"/>
N° de Identificación	<input type="text"/>

Formulario Ingresar Cobertura

Código de la Póliza	<input type="text"/>
Cobertura	<input type="text"/>
Monto a Pagar	<input type="text"/>

Formulario Configuración de Cuota

Código de Póliza	<input type="text"/>
Forma de Pago	<input type="text"/>
Valor de la Cuota	<input type="text"/>

Formulario Cotizaciones

N° de Cotización	<input type="text" value="Autonum"/>
Póliza	<input type="text"/>
En Atención a	<input type="text"/>
Promotor	<input type="text"/>

Formulario Forma de Pago

Código	<input type="text" value="Autonumérico"/>
Descripción	<input type="text"/>
N° de Pagos	<input type="text"/>

Formulario Información de la Empresa

Empresa	<input type="text"/>
Dirección	<input type="text"/>
Teléfono	<input type="text"/>
N° RUC	<input type="text"/>



Formulario Ingresar Tasa de Cambio

Tasa del día	<input type="text"/>
Tasa de Cambio	<input type="text"/>



Formulario Compra de Póliza

Proveedor	<input type="text"/>
Código	<input type="text"/>
Descripción	<input type="text"/>
Valor de Compra	<input type="text"/>
N° de Beneficiarios	<input type="text"/>
Valor de Venta	<input type="text"/>
% de Comisión	<input type="text"/>
 	

Formulario Ingreso de Promotor

Código del Promotor	<input type="text" value="Autonum"/>
Nombre	<input type="text"/>
Dirección	<input type="text"/>
Teléfono	<input type="text"/>
Celular	<input type="text"/>
Fecha de Ingreso	<input type="text"/>
Salario	<input type="text"/>
 	

Formulario Ingreso de Proveedor

Código	<input type="text" value="Autonum"/>	Proveedor	<input type="text"/>
Dirección	<input type="text"/>		
Contacto	<input type="text"/>		
Teléfono	<input type="text"/>	Extención	<input type="text"/>
 			

Formulario Ingreso de Tipo de Identificación

Código	<input type="text" value="Autonómico"/>
Identificación	<input type="text"/>
 	

Formulario Ingreso de Modalidad de Pago

Código	<input type="text" value="Autonómico"/>
Modalidad de Pago	<input type="text"/>
 	

Formulario Ingreso de Venta

N° de Venta	<input type="text" value="Autonómico"/>
Fecha de Adquisición	<input type="text"/>
Cliente	<input type="text"/>
Póliza	<input type="text"/>
Promotor	<input type="text"/>
Forma de Pago	<input type="text"/>
 	

Formulario Agregar Beneficiario

N° de Venta	<input type="text"/>
Beneficiario	<input type="text"/>
 	

Formulario Agregar Cobertura

Código de la Póliza	<input type="text"/>
Cobertura	<input type="text"/>
Valor de la Cobertura	<input type="text"/>
 	

Formulario Agregar

N° de la Póliza	<input type="text"/>
Forma de Pago	<input type="text"/>
Valor de la Cuota	<input type="text"/>
 	

Formulario Modificar Banco

Buscar Banco	<input type="text"/>
Código	<input type="text" value="(Autonómico)"/>
Banco	<input type="text"/>
 	

Formulario Modificar y Eliminar Beneficiario

N° de la Venta	<input type="text"/>	Beneficiario	<input type="text"/>
Beneficiario	<input type="text"/>		
  			

Formulario Modificar Cliente

Buscar Cliente	<input type="text"/>
Código del Cliente	<input type="text" value="(Autonómico)"/>
Fecha de Visita	<input type="text"/>
Nombre	<input type="text"/>
Dirección	<input type="text"/>
Teléfono	<input type="text"/>
Celular	<input type="text"/>
Identificación	<input type="text"/>
N° de Identificación	<input type="text"/>

Formulario Modificar y Eliminar Cobertura

Código de la Póliza	<input type="text"/>
Elegir Cobertura	<input type="text"/>
Cobertura	<input type="text"/>
Valor de la Cobertura	<input type="text"/>

Formulario Modificar Forma de Pago

Buscar Forma de Pago	<input type="text"/>
Código	<input type="text" value="(Autonómico)"/>
Descripción	<input type="text"/>
N° de Pagos	<input type="text"/>

Formulario Modificar Póliza

Código	<input type="text"/>
Proveedor	<input type="text"/>
Descripción	<input type="text"/>
Valor de Compra	<input type="text"/>
N° de Beneficiarios	<input type="text"/>
Valor de Venta	<input type="text"/>
% de Comisión	<input type="text"/>
  	

Formulario Modificar Promotor

Elegir Promotor	<input type="text"/>
Código del Promotor	<input type="text" value="(Autonumérico)"/>
Nombre	<input type="text"/>
Dirección	<input type="text"/>
Teléfono	<input type="text"/>
Celular	<input type="text"/>
Fecha de Ingreso	<input type="text"/>
Salario	<input type="text"/>
  	

Formulario Modificar Proveedor

Elegir Proveedor	<input type="text"/>	
Código	<input type="text" value="numérico"/>	Proveedor <input type="text"/>
Dirección	<input type="text"/>	
Contacto	<input type="text"/>	
Teléfono	<input type="text"/>	Extensión <input type="text"/>
  		

Formulario Modificar Tipo de Identificación

Buscar Identificación	<input type="text"/>
Código	<input type="text" value="(Autonumérico)"/>
Identificación	<input type="text"/>
  	

Formulario Modificar Tipo de Pago

Modalidad de Pago	<input type="text"/>
Código	<input type="text" value="(Autonumérico)"/>
Modalidad de Pago	<input type="text"/>
  	

Formulario Modificar Venta

N° de la Venta	<input type="text"/>
Fecha de Adquisición	<input type="text"/>
Cliente	<input type="text"/>
Póliza	<input type="text"/>
Promotor	<input type="text"/>
Forma de Pago	<input type="text"/>

D- Formatos de Salida

Las salidas del sistema serán a través de reportes, los cuales se describen a continuación:

Reporte Clientes Activos

Corredores de Seguros Generales S.A				
Shell Ciudad Jardín 1/2 c. al sur				
RUC 1548-89721-12			Teléfono 278-34-70	
Clientes Activos				
Código	Nombre	Visitad	Dirección	Teléfono
<hr/>				

Reporte Clientes Inactivos

Corredores de Seguros Generales S.A

Shell Ciudad Jardin 1/2 c. al sur

RUC 1548-89721-12

Teléfono 278-34-70

Clientes Inactivos

<u>Código</u>	<u>Nombre</u>	<u>Visitado</u>	<u>Dirección</u>	<u>Teléfono</u>
---------------	---------------	-----------------	------------------	-----------------

Reporte Pago de Comisiones

Corredores de Seguros Generales S.A

Shell Ciudad Jardin 1/2 c. al sur

RUC 1548-89721-12

Teléfono 278-34-70

Pago de Comisiones

Promotor

<u>N° de Recibo</u>	<u>Valor</u>	<u>Porcentaje</u>	<u>Fecha de Cancelació</u>	<u>Comisión</u>
---------------------	--------------	-------------------	----------------------------	-----------------

Neto a Pagar

C\$ #Error

Total de Comisiones

C\$ #Error

Reporte Estado de Cuenta por Venta

Corredores de Seguros Generales S.A

Shell Ciudad Jardin 1/2 c. al sur

RUC 1548-89721-12

Teléfono 278-34-70

Estado de Cuenta por Venta

ID_VENTA

NOMBRE

DESCRIPCION

ID_RECIBO	ALOR_RECIBO	DEL	AL ESTADO	FECHA_CANCEL
-----------	-------------	-----	-----------	--------------

Reporte Aviso de Cobro

Corredores de Seguros Generales S.A

Shell Ciudad Jardin 1/2 c. al sur

RUC 1548-89721-12

Teléfono 278-34-70

Aviso de Cobro

NOMBRE

DIRECCION

TELEFONO

CELULAR

DESCRIPCION

ESTADO

Nº de Recibo	Valor	Del	Al	Fecha de Cancelación	Fecha de Emisión
--------------	-------	-----	----	----------------------	------------------

Reporte Recibo Oficial

Corredores de Seguros Generales S.A			
Shell Ciudad Jardín 1/2 c. al sur		Teléfono 278-34-70	
RUC 1548-89721-12			
RECIBO OFICIAL			
POLIZA			
NOMBRE			
DIRECCION			
TELEFONO		CELULAR	
VIGENCIA	Del	(0001 A.M) Al	(2400 P.M)
FECHA DE EMISION			
Número de Recibo		Valor del Recibo U\$	
Tipo de Pago		Fecha de Cancelación	
Institución Financiera		Referencia	
Moneda Extranjera U\$		Tasa de Cambio	
Moneda Nacional C\$			
		Sub-Total C\$	
		IGV	
		Total a Pagar C\$	
En concepto de pago correspondiente al período: /			
Valor en Letras			

Reporte Estado de Mora

Corredores de Seguros Generales S.A			
Shell Ciudad Jardín 1/2 c. al sur		Teléfono 278-34-70	
RUC 1548-89721-12			
Estado de Mora			
NOMBRE			
DIRECCION			
	<u>Nº de Recibo</u>	<u>Valor del Recibo</u>	<u>Fecha de Emisión</u>
Mora	C\$	#Error	
Mora Total	C\$	#Error	

Reporte Ventas por Fecha

Corredores de Seguros Generales S.A

Shell Ciudad Jardin 1/2 c. al sur

RUC 1548-89721-12

Teléfono 278-34-70

Ventas por Fecha

Venta	Póliza	Adquirida	Promotor
--------------	---------------	------------------	-----------------

Reporte Ventas por Póliza

Corredores de Seguros Generales S.A

Shell Ciudad Jardin 1/2 c. al sur

RUC 1548-89721-12

Teléfono 278-34-70

Ventas por Póliza

Gráfica de Ventas por Póliza

Reporte Ventas por Promotor

Corredores de Seguros Generales S.A.	
Shell Ciudad Jardin 1/2 c. al sur RUC 1548-89721-12	Teléfono 278-34-70
<h3>Ventas por Promotor</h3>	
<div style="border: 1px solid black; padding: 20px;"><h3>Gráfica de Ventas por Promotor</h3></div>	

Reporte Ventas por Vencerse

Corredores de Seguros Generales S.A.				
Shell Ciudad Jardin 1/2 c. al sur RUC 1548-89721-12		Teléfono 278-34-70		
<h3>Ventas por Vencerse</h3>				
Venta	Nombre	Dirección	Teléfono	Celular
<hr/>				
Fecha de Vencimiento				

VII- CONCLUSIÓN

Al finalizar el proyecto, podemos asegurar que el Sistema de Administración de Ventas de Seguros satisface los requerimientos planteados por la institución, lo que nos lleva a obtener los beneficios esperados.

Estos beneficios se traducen en un decremento en la carga laboral de los dos empleados de administración, los cuales se encargaban de mantener al día los estados de cuenta, mora y cobros que se debían elaborar. Dicho decremento le permitirá a la institución dejar estas labores a un solo empleado a cargo del sistema para que el otro se incluya en el equipo de ventas y de esta manera elevarlas entre el 15% y el 20%.

VIII- RECOMENDACIONES

- Se recomienda a Corredores de Seguros Generales S. A. capacitar a la persona que usara el sistema, debido a que es un personal pequeño y no incurre en los costos.
- Dar mantenimiento constante al equipo y al sistema, para mantener su rendimiento.
- Ubicar la computadora en lugares libres de polvo, con una temperatura adecuada para que este equipo trabaje adecuadamente de acuerdo a sus especificaciones.
- Se sugiere el uso del Antivirus Norton debido a su amplia lista de virus que es capaz de detectar con respecto a otros tipos de Antivirus, se sugiere que sea actualizado para evitar cualquier daño a la información o al equipo por el surgimiento de algún nuevo virus.
- Hacer respaldos en disquetes de la información por lo menos cada semana.

ANEXOS

INDICE DE TABLAS	PÁG.
Tabla VIII.1: Actividades a realizar en el desarrollo de la Aplicación	46
Tabla VIII.2- Factibilidad Económica de la Alternativa I	49
Tabla VIII.3: Factibilidad Económica de la Alternativa II	52
Tabla VIII.4: Datos contables de la Empresa	53
Tabla VIII.5: Datos proyectados al termino de un año	54
Tabla VIII.6: Datos proyectados al termino de un año	54
Tabla X.1: Barra de Menú Principal del Sistema	86

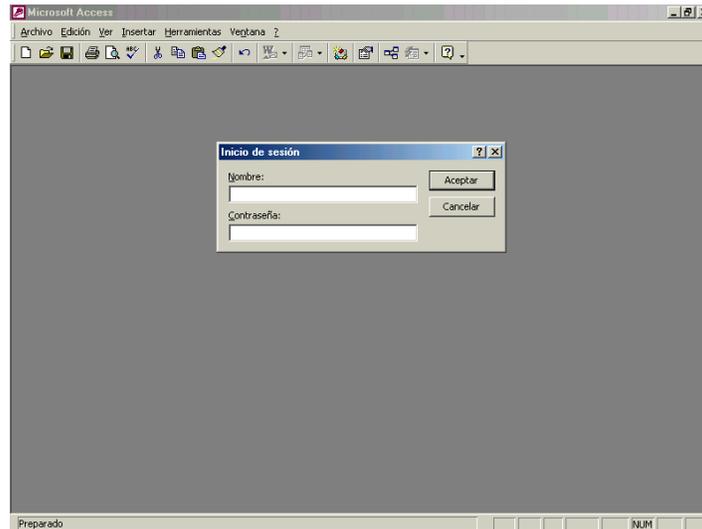
INDICE DE FIGURAS	PÁG.
Figura V.1: El Ciclo de Vida Clásico según Roger S. Pressman.	13
Figura V.2: Modelo de Desarrollo Incremental.	15
Figura V.3: Prototipo de Requerimientos basado en el modelo de desarrollo incremental	16
Figura VII.1: Organigrama de la Empresa	42
Figura IX.1: Diagrama de Contexto (Nivel 0)	56
Figura IX.2: Diagrama Nivel 1	57
Figura IX.3: Diagrama Nivel 2	58
Figura IX.4: Diagrama Nivel 3	59
Figura X.1: Diagrama Entidad / Relación del Sistema	88

MANUAL DE USUARIO

Ver archivo Manual de Usuario.pdf

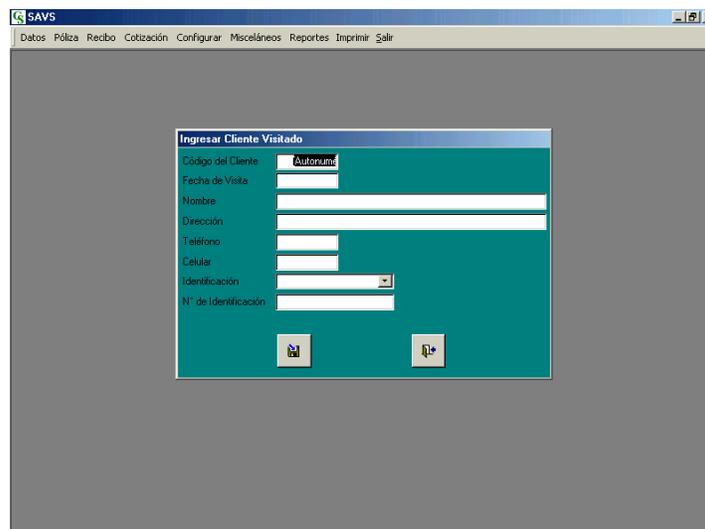
MANUAL DE USUARIO

Inicio



Esta es la ventana inicial del sistema, la cual presenta la ventana de Inicio de Sesión en la cual el usuario deberá introducir su nombre de usuario y su contraseña. Una vez registrado en el sistema se presentará la barra de menú principal que contiene las diversas opciones que contiene el sistema. En el presente manual se explicará cada una de estas opciones.

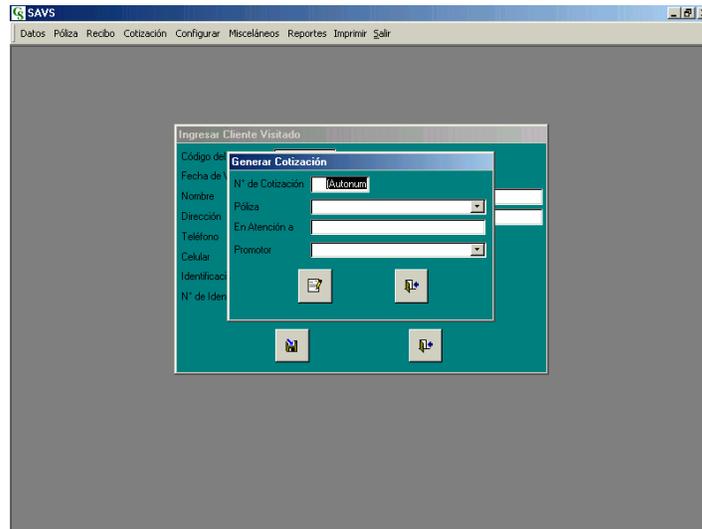
Ingresar Cliente



Esta opción se encuentra en **Datos-Clientes-Ingresar**, cada uno de estos campos deberán de ser llenados con la información que se solicita, con excepción de *Teléfono* y

Celular los cuales son opcionales. Además de los campos también existe el botón Guardar y el botón Salir.

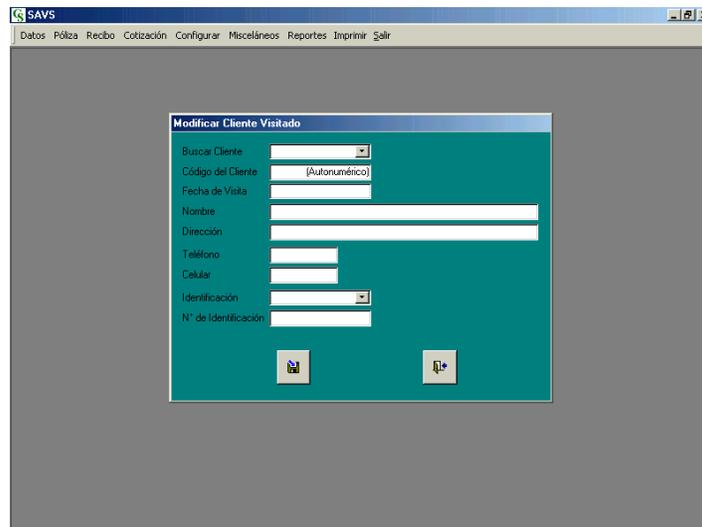
Botón Guardar: Guarda los datos si los campos requeridos contienen la información debida y pregunta si desea elaborar una cotización, cuando la respuesta es *Si* aparecerá la siguiente ventana:



En este formulario se ingresarán todos los datos solicitados. El botón *Generar Reporte* ejecuta la cotización y el botón *Salir* cierra el formulario de Cotizaciones.

Botón Salir: Cierra el formulario.

Modificar Cliente

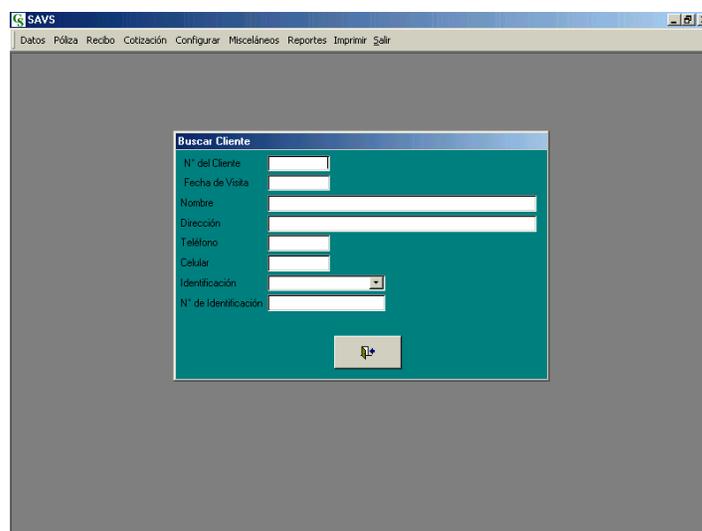


Esta opción se encuentra en **Datos-Clientes-Modificar**, para buscar al cliente a modificar, este debe ser seleccionado del campo *Buscar Cliente* o puede ser escrito en el mismo. El único campo que no puede ser modificado es *Código del Cliente*.

Botón Guardar: Guarda los cambios si el nombre del cliente no se repite con los antes guardados.

Botón Salir: Cierra el formulario.

Visualizar Cliente



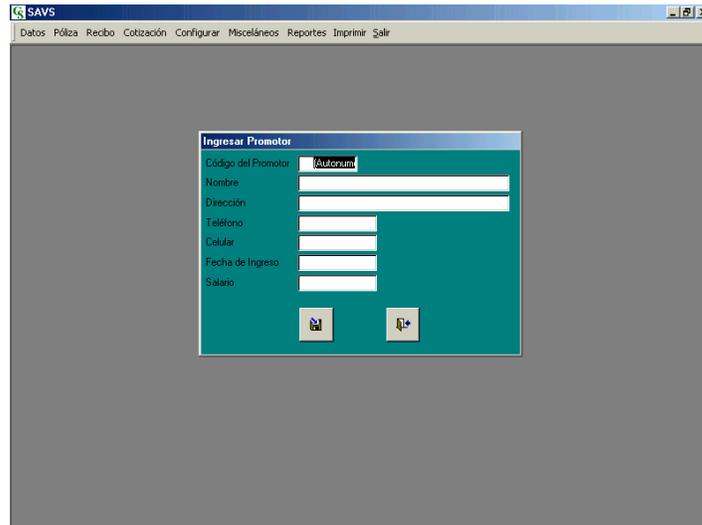
The image shows a screenshot of a software application window titled 'SAVS'. The window has a menu bar with the following items: 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. The main content area is a form titled 'Buscar Cliente' with a teal background. The form contains the following fields and controls:

- 'N° del Cliente': A text input field.
- 'Fecha de Visita': A date selection field.
- 'Nombre': A text input field.
- 'Dirección': A text input field.
- 'Teléfono': A text input field.
- 'Celular': A text input field.
- 'Identificación': A dropdown menu.
- 'N° de Identificación': A text input field.
- A 'Buscar' button with a magnifying glass icon at the bottom right of the form.

Esta opción se encuentra en **Datos-Clientes-Visualizar**, en el campo *N° del Cliente* se ingresa el número del cliente del cual se desea visualizar los datos.

Botón Salir: Cierra el formulario.

Ingresar Promotor



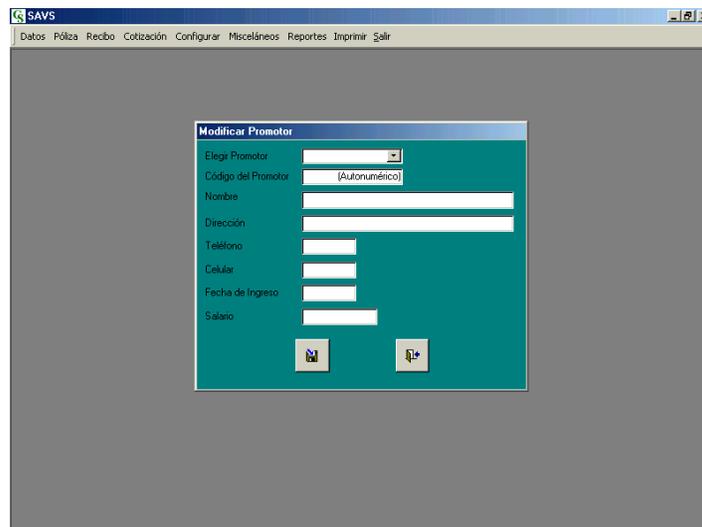
The screenshot shows a window titled 'SAVS' with a menu bar containing 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. The main content area displays a form titled 'Ingresar Promotor' with the following fields: 'Código del Promotor' (with a dropdown menu showing '(Autonumérico)'), 'Nombre', 'Dirección', 'Teléfono', 'Celular', 'Fecha de Ingreso', and 'Salario'. At the bottom of the form are two buttons: 'Guardar' and 'Salir'.

Esta opción se encuentra en **Datos-Promotores-Ingresar**, cada uno de estos campos deberán de ser llenados con la información que se solicita, con excepción de *Teléfono* y *Celular* los cuales son opcionales. Además de los campos también existe el botón Guardar y el botón Salir.

Botón Guardar: Guarda los datos si los campos requeridos contienen la información debida.

Botón Salir: Cierra el formulario.

Modificar Promotor



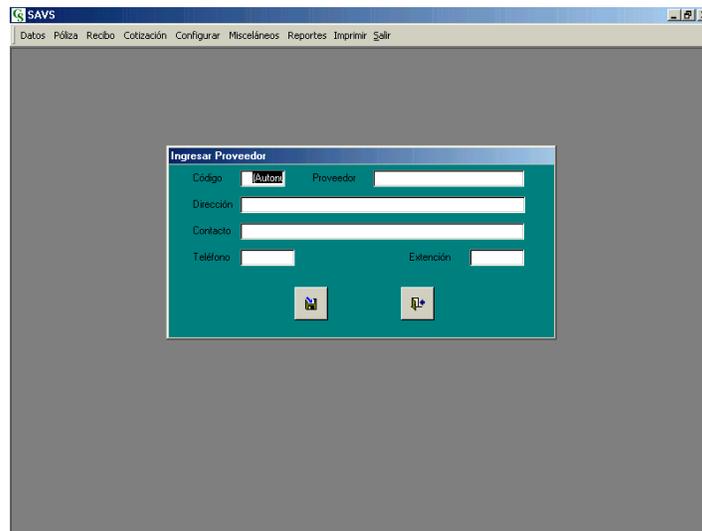
The screenshot shows a window titled 'SAVS' with a menu bar containing 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. The main content area displays a form titled 'Modificar Promotor' with the following fields: 'Elegir Promotor' (with a dropdown menu), 'Código del Promotor' (with a dropdown menu showing '(Autonumérico)'), 'Nombre', 'Dirección', 'Teléfono', 'Celular', 'Fecha de Ingreso', and 'Salario'. At the bottom of the form are two buttons: 'Guardar' and 'Salir'.

Esta opción se encuentra en **Datos-Promotores-Modificar**, para buscar al promotor a modificar, este debe ser seleccionado del campo *Elegir Promotor*. El único campo que no puede ser modificado es *Código del Promotor*.

Botón Guardar: Guarda los cambios si el nombre del promotor no se repite con los antes guardados.

Botón Salir: Cierra el formulario.

Ingresar Proveedor



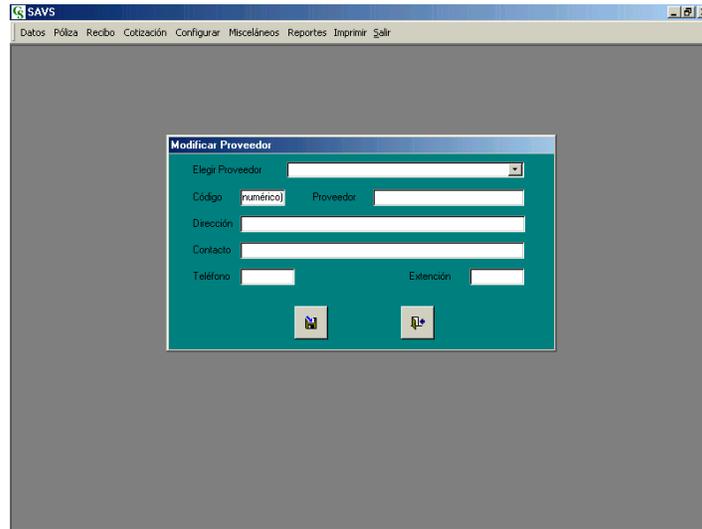
The image shows a screenshot of a web application window titled 'SAVS'. The main menu includes 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. A central dialog box titled 'Ingresar Proveedor' is displayed. This dialog box contains the following fields: 'Código' (with a dropdown menu), 'Proveedor' (text input), 'Dirección' (text input), 'Contacto' (text input), 'Teléfono' (text input), and 'Extensión' (text input). At the bottom of the dialog box, there are two buttons: a 'Guardar' button (represented by a floppy disk icon) and a 'Salir' button (represented by a square icon with a diagonal line).

Esta opción se encuentra en **Datos-Proveedores-Ingresar**, cada uno de estos campos deberán de ser llenados con la información que se solicita, con excepción de *Extensión* el cual es opcional. Además de los campos también existe el botón Guardar y el botón Salir.

Botón Guardar: Guarda los datos si los campos requeridos contienen la información debida.

Botón Salir: Cierra el formulario.

Modificar Proveedor



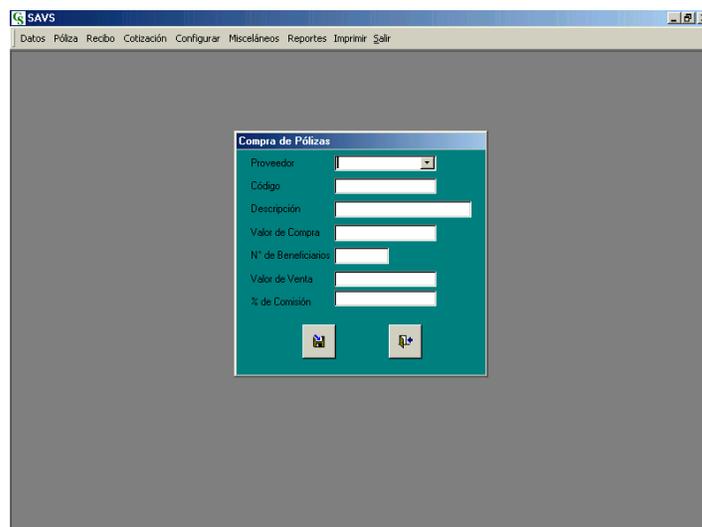
The screenshot shows a web application window titled 'SAVS' with a menu bar containing 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. The main content area displays a modal form titled 'Modificar Proveedor'. The form includes the following fields: 'Elegir Proveedor' (a dropdown menu), 'Código' (a text input with 'numérico' in a small box next to it), 'Proveedor' (a text input), 'Dirección' (a text input), 'Contacto' (a text input), 'Teléfono' (a text input), and 'Extensión' (a text input). At the bottom of the form are two buttons: a 'Guardar' button with a floppy disk icon and a 'Salir' button with a right-pointing arrow icon.

Esta opción se encuentra en **Datos-Proveedores-Modificar**, para buscar al proveedor a modificar, este debe ser seleccionado del campo *Elegir Proveedor*. El único campo que no puede ser modificado es *Código del Proveedor*.

Botón Guardar: Guarda los cambios si el nombre del proveedor no se repite con los antes guardados.

Botón Salir: Cierra el formulario.

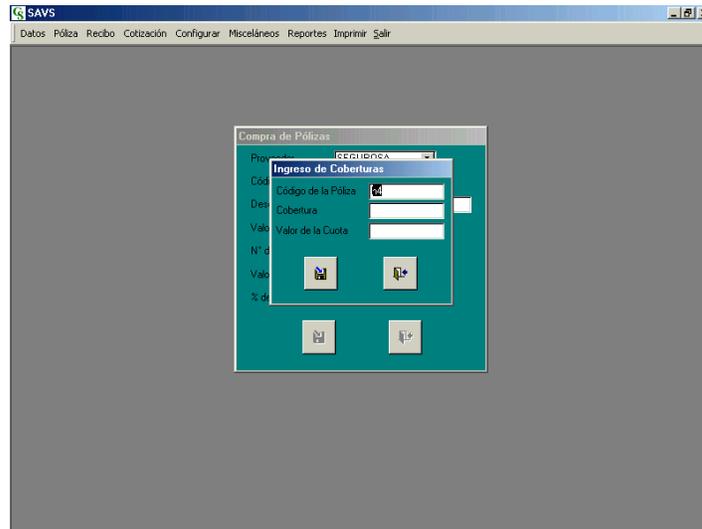
Ingresar Póliza



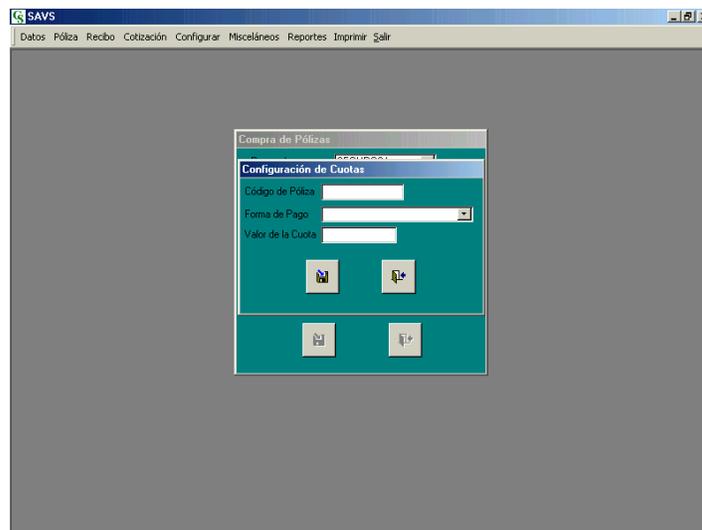
The screenshot shows a web application window titled 'SAVS' with a menu bar containing 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. The main content area displays a modal form titled 'Compra de Pólizas'. The form includes the following fields: 'Proveedor' (a dropdown menu), 'Código' (a text input), 'Descripción' (a text input), 'Valor de Compra' (a text input), 'N° de Beneficiarios' (a text input), 'Valor de Venta' (a text input), and '% de Comisión' (a text input). At the bottom of the form are two buttons: a 'Guardar' button with a floppy disk icon and a 'Salir' button with a right-pointing arrow icon.

Esta opción se encuentra en **Póliza-Comprar-Ingresar**, cada uno de estos campos deberán de ser llenados con la información que se solicita. Además de los campos también existe el botón Guardar y el botón Salir.

Botón Guardar: Guarda los datos si todos los campos contienen la información debida y abre la ventana de ingreso de Coberturas, la cual es la siguiente:



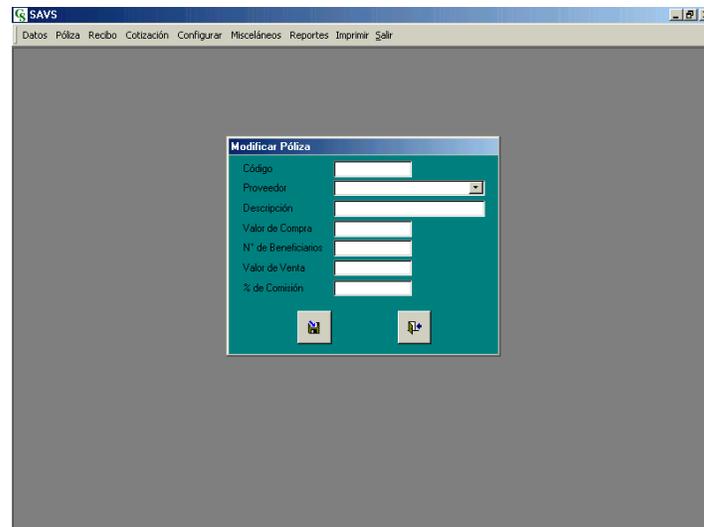
Aquí se ingresan las coberturas de la póliza, todos estos campos son requeridos. El botón Guardar guarda la cobertura y el botón Salir cierra la ventana *Ingreso de Coberturas* y abre la ventana *Configuración de Cuotas*, la cual se muestra a continuación:



Aquí se ingresan las cuotas de la póliza, al elegir la forma de pago aparecerá el código de la póliza, todos estos campos son requeridos. El botón Guardar guarda la configuración de la cuota y el botón Salir cierra la ventana *Configuración de Cuotas*.

Botón Salir: Cierra el formulario.

Modificar Póliza



The image shows a screenshot of a software application window titled 'SAVS'. The main menu bar includes 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. A modal dialog box titled 'Modificar Póliza' is open, containing the following fields and controls:

- Código: Text input field.
- Proveedor: Dropdown menu.
- Descripción: Text input field.
- Valor de Compra: Text input field.
- N° de Beneficiarios: Text input field.
- Valor de Venta: Text input field.
- % de Comisión: Text input field.

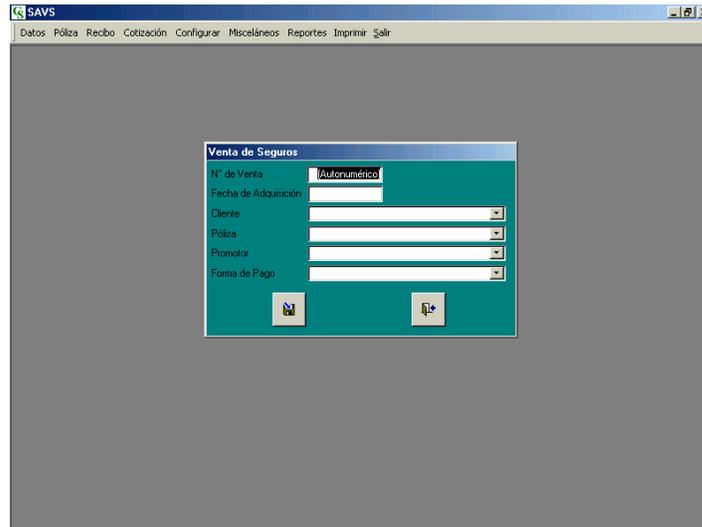
At the bottom of the dialog box, there are two buttons: a 'Guardar' button (represented by a floppy disk icon) and a 'Salir' button (represented by a door icon).

Esta opción se encuentra en **Póliza-Comprar-Modificar**, para buscar la póliza a modificar, este debe ser escrito en el campo *Código*. El único campo que puede ser modificado es *Descripción*.

Botón Guardar: Guarda los cambios.

Botón Salir: Cierra el formulario.

Ingresar Venta



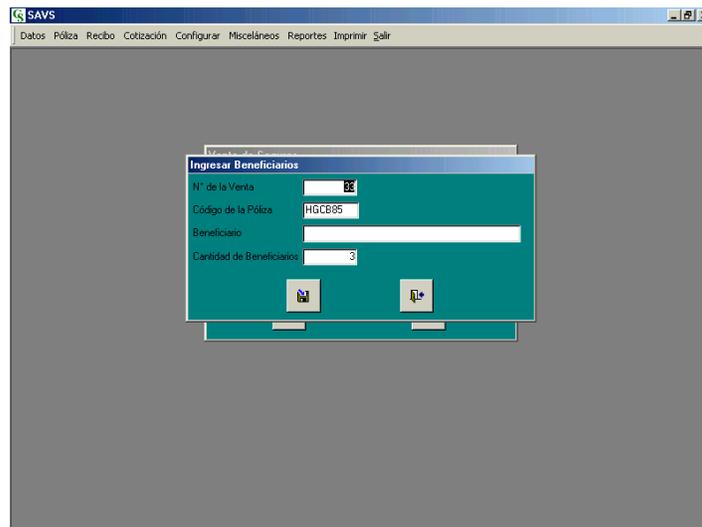
The screenshot shows a window titled 'SAVS' with a menu bar containing 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. The main content area displays a form titled 'Venta de Seguros' with the following fields:

- N° de Venta:
- Fecha de Adquisición:
- Cliente:
- Póliza:
- Promotor:
- Forma de Pago:

At the bottom of the form are two buttons: a 'Guardar' button (represented by a floppy disk icon) and a 'Salir' button (represented by a door icon).

Esta opción se encuentra en **Póliza-Vender-Ingresar**, cada uno de estos campos deberán de ser llenados con la información que se solicita. Además de los campos también existe el botón Guardar y el botón Salir.

Botón Guardar: Guarda los datos si los campos requeridos contienen la información debida y abre la ventana *Ingresar Beneficiarios*, el cual se muestra a continuación:



The screenshot shows the same 'SAVS' window with the 'Ingresar Beneficiarios' form displayed. The form contains the following fields:

- N° de la Venta:
- Código de la Póliza:
- Beneficiario:
- Cantidad de Beneficiarios:

At the bottom of the form are two buttons: a 'Guardar' button (represented by a floppy disk icon) and a 'Salir' button (represented by a door icon).

Aquí se debe ingresar los beneficiarios de la venta. El Botón guardar, guarda los datos del beneficiario y el Botón Salir cierra la ventana de *Ingresar Beneficiarios* y abre la ventana de *Cancelar Recibo*, la cual se presenta a continuación:

En esta ventana se ingresan los datos de la cancelación del primer Recibo. El Botón Cancelar Recibo guarda los datos de la cancelación y cierra la ventana.

Botón Salir: Cierra el formulario.

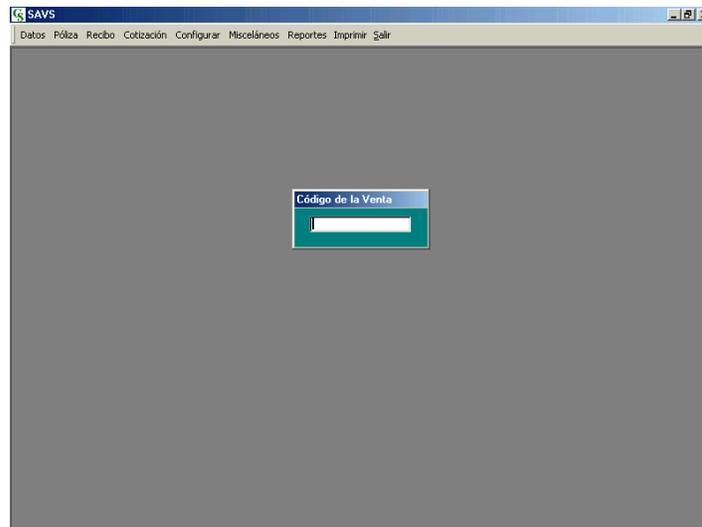
Modificar Venta

Esta opción se encuentra en **Póliza-Vender-Modificar**, para buscar la venta a modificar, este debe ser escrito en el campo *N° de Venta*. Los únicos campos que pueden ser modificados es *Promotor* y *forma de pago*.

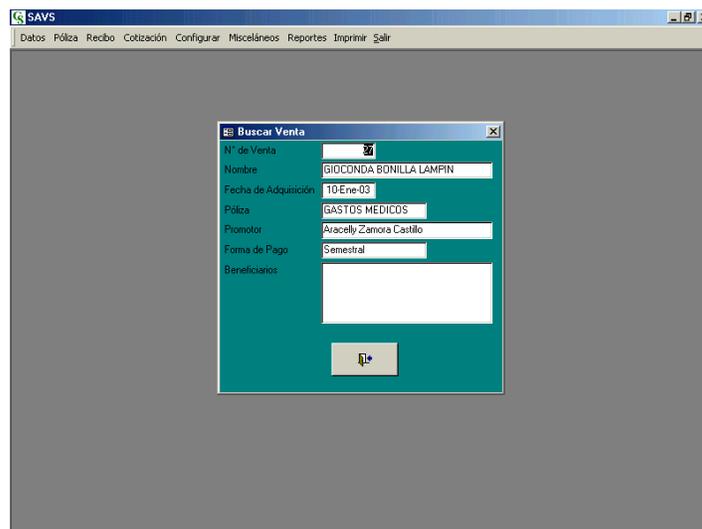
Botón Guardar: Guarda los cambios.

Botón Salir: Cierra el formulario.

Visualizar Venta



Esta opción se encuentra en **Póliza-Vender-Visualizar**, en el campo se ingresa el número de la venta del cual se desea visualizar los datos. Una vez ingresado el número aparecerá la siguiente ventana:



Esta ventana únicamente es de visualización por lo tanto el único botón es el de Salir, el cual cierra el formulario.

Cancelar Recibo

The screenshot shows a web application window titled 'SAVS' with a menu bar containing 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. The main content area displays a form titled 'Cancelar Recibo'. The form includes the following fields:

- N° de Recibo: Text input field
- Nombre: Text input field
- Total en Dolares: Text input field
- Modalidad de Pago: Dropdown menu
- Total en Cordobas: Text input field
- Banco: Dropdown menu
- Referencia: Text input field

Below these fields is a calculation section:

Total Recibo = (Dolares x Tasa) + Cordobas

Each part of the calculation has a corresponding text input field. At the bottom of the form is a 'Valor en Letras' text input field. Two buttons are located at the bottom: a red house icon and a button with a right-pointing arrow and a plus sign.

Esta opción se encuentra en **Recibo-Cancelar**. En el campo *N° de Recibo* se ingresa el número del recibo el cual se desea cancelar, los campos necesarios van a depender de la modalidad de pago.

Botón Cancelar Recibo: Guarda los datos de la cancelación del recibo.

Botón Salir: Cierra el formulario

Imprimir Recibos

The screenshot shows the same 'SAVS' application window. The main content area displays a form titled 'Impresión de Recibos'. The form includes the following elements:

- A section titled 'Imprimir Avisos de Cobro' with two radio buttons: 'Por Número' (selected) and 'Por Fecha de Emisión'.
- A 'Cobro Hº' text input field.
- A 'Del' text input field.
- A 'Al' text input field.

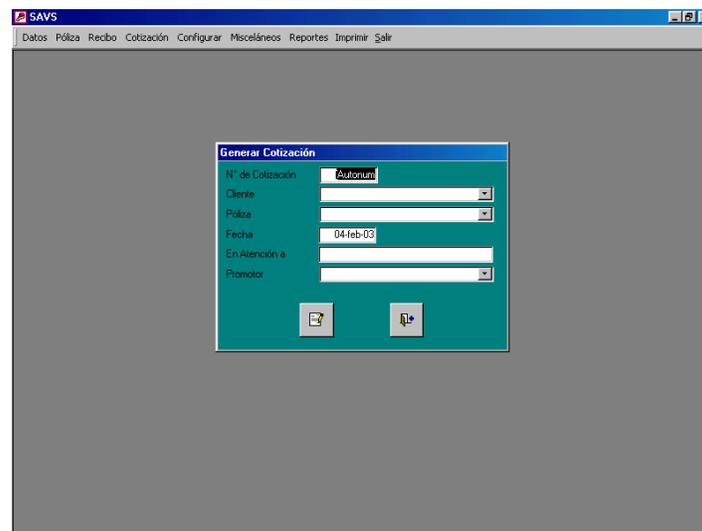
At the bottom of the form are two buttons: a printer icon and a button with a right-pointing arrow and a plus sign.

Esta opción se encuentra en **Recibo-Imprimir**, la cual nos permite imprimir los recibos ya sea por el número de recibo o por fecha de emisión. Si se escoge por número se debe ingresar el número en el campo *Recibo N°* y si se escoge por fecha se debe ingresar el rango de fecha en los campos *del* y *al*.

Botón Generar: Realiza los reportes según los parámetros introducidos.

Botón Salir: Cierra el formulario.

Generar Cotización



The image shows a screenshot of a software application window titled 'SAVS'. The menu bar includes 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. The main content area displays a form titled 'Generar Cotización' with the following fields:

- N° de Cotización:
- Cliente:
- Póliza:
- Fecha:
- En Atención a:
- Promotor:

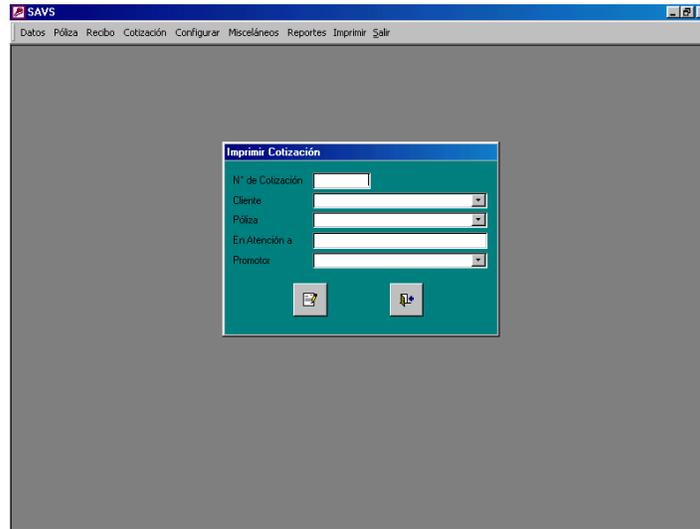
At the bottom of the form are two buttons: a 'Generar' button (represented by a document icon) and a 'Salir' button (represented by a right-pointing arrow icon).

Esta opción se encuentra en **Cotización-Generar**. Para generar una cotización se deben llenar los datos solicitados con excepción del campo *En Atención A* el cual es opcional.

Botón Generar: Genera la cotización con los datos introducidos.

Botón Salir: Cierra el formulario.

Imprimir Cotización

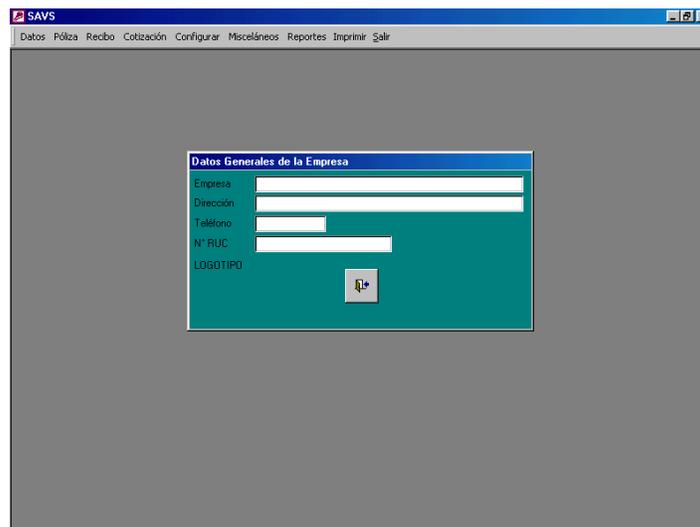


Esta opción se encuentra en **Cotización-Imprimir**. Para visualizar antes de imprimir una cotización se deberá introducir el numero de la cotización en el campo *N de Cotización*.

Botón imprimir: Imprime la cotización activa.

Botón Salir: Cierra el formulario.

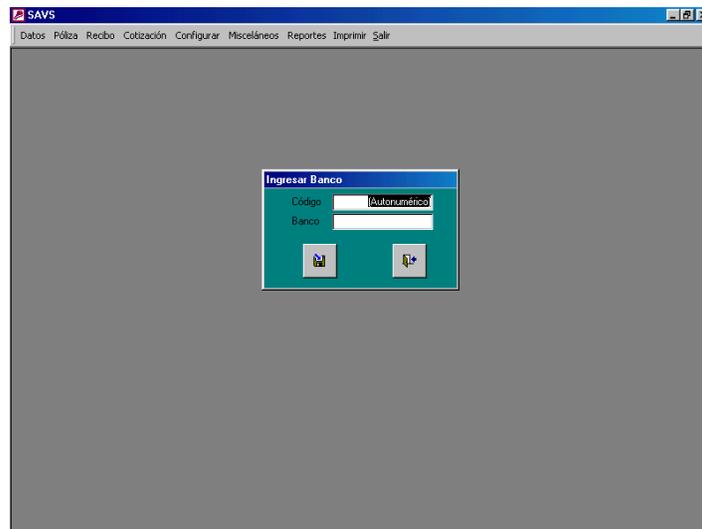
Introducir Datos de la Empresa



Esta opción se encuentra en **Configurar-Empresa**. Aquí se deben ingresar los datos solicitados en esta ventana.

Botón Salir: cierra el formulario.

Ingresar Banco



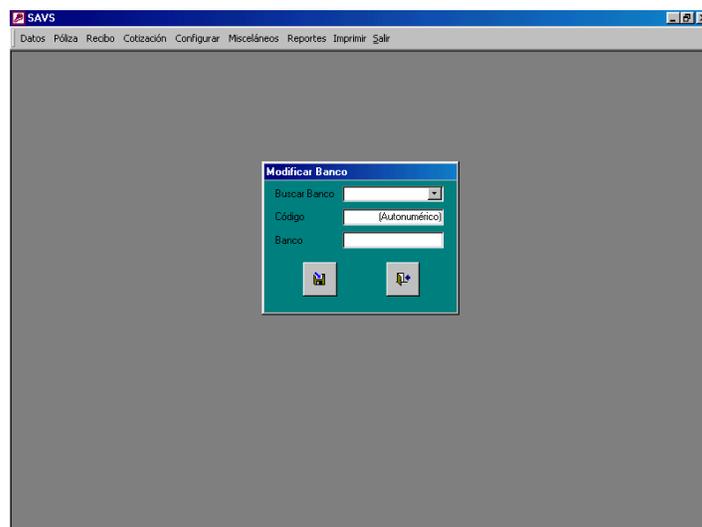
The screenshot shows a window titled 'SAVS' with a menu bar containing 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. In the center, a smaller dialog box titled 'Ingresar Banco' is displayed. This dialog box has two text input fields: 'Código' with the placeholder text '[Autonumérico]' and 'Banco'. Below the fields are two buttons: a 'Guardar' button (represented by a floppy disk icon) and a 'Salir' button (represented by a mouse cursor icon).

Esta opción se puede encontrar en **Configurar-Banco-Ingresar**.

Botón Guardar: Guarda el banco siempre y cuando no se repita con uno antes guardado.

Botón Salir: Cierra el formulario

Modificar Banco



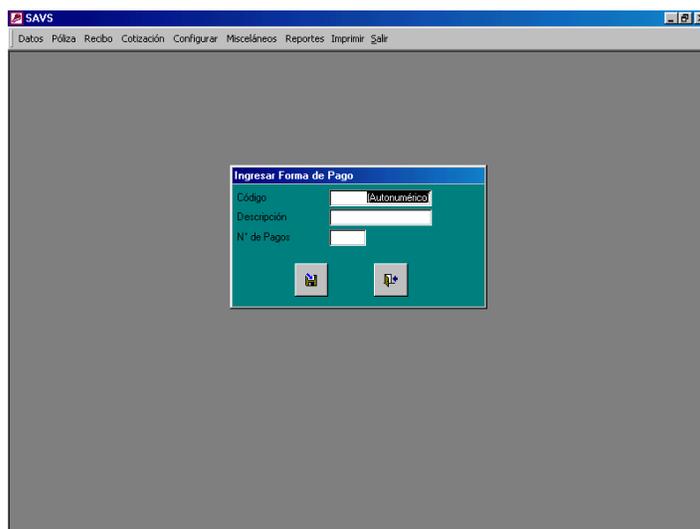
The screenshot shows the same 'SAVS' application window. In the center, a dialog box titled 'Modificar Banco' is displayed. This dialog box includes a dropdown menu labeled 'Buscar Banco', a 'Código' text input field with the placeholder '[Autonumérico]', and a 'Banco' text input field. At the bottom, there are two buttons: a 'Guardar' button (floppy disk icon) and a 'Salir' button (mouse cursor icon).

Esta opción se encuentra en **Configurar-Banco-Modificar**. Se debe buscar el banco a modificar en el campo *Buscar Banco*. El campo *código* no puede ser modificado.

Botón Guardar: Guarda los cambios realizados, siempre y cuando el banco no se repita.

Botón Salir: Cierra el formulario.

Ingresar Forma de Pago



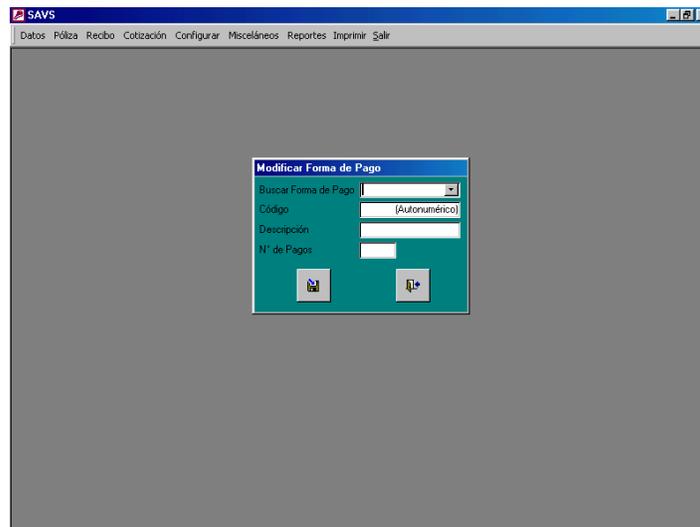
The image shows a screenshot of a software application window titled 'SAVS'. The main menu bar includes 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. The central area of the window displays a smaller window titled 'Ingresar Forma de Pago'. This sub-window contains three input fields: 'Código' with the value 'Autonómico', 'Descripción', and 'N° de Pagos'. Below the fields are two buttons: a 'Guardar' button (represented by a floppy disk icon) and a 'Salir' button (represented by a door icon).

Esta opción se encuentra en **Configurar-Forma de Pago-Ingresar**. Para ingresar una forma de pago se debe llenar los campos presente en la ventana anterior. El campo *N de Pagos* debe ser llenado de la siguiente manera. Se divide 12(total de mes en el año) entre la cantidad de meses que tiene la forma de pago (Mensual =1, Trimestral =3, Cuatrimestral =4, Semestral 6, Contado = 12), dando como resultado el numero de pago.

Botón Guardar: Guarda los datos de la forma de Pago si no se repite con una antes guardada.

Botón Salir: Cierra el formulario.

Modificar Forma de Pago



The screenshot shows a window titled 'SAVS' with a menu bar containing 'Datos', 'Póliza', 'Recibo', 'Cobización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. In the center, a dialog box titled 'Modificar Forma de Pago' is displayed. It contains the following fields and controls:

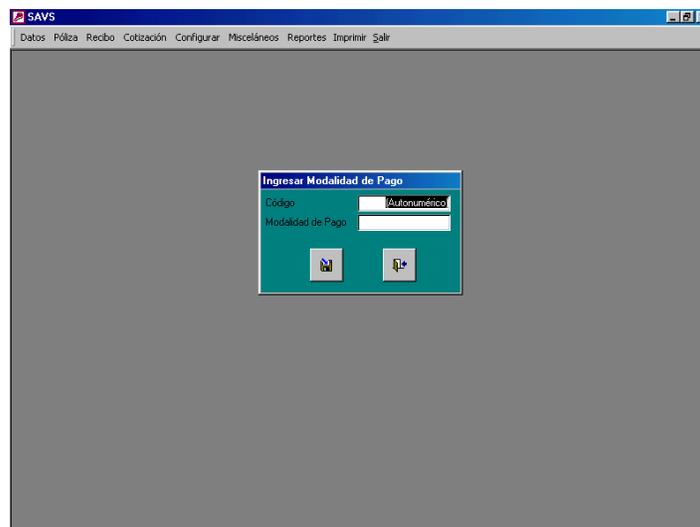
- 'Buscar Forma de Pago': A dropdown menu.
- 'Código': A text field with '(Autonumérico)' displayed inside.
- 'Descripción': A text field.
- 'N° de Pagos': A text field.
- Two buttons at the bottom: a 'Guardar' (Save) button and a 'Salir' (Exit) button.

Esta opción se encuentra en **Configurar-Forma de Pago-Modificar**. Para Buscar una forma de pago se debe buscar en el campo *Buscar Forma de Pago*. El campo *Código* no puede ser modificado.

Botón Guardar: Guarda los datos de la forma de Pago si no se repite con una antes guardada.

Botón Salir: Cierra el formulario.

Ingresar Modalidad de Pago



The screenshot shows the same 'SAVS' window. In the center, a dialog box titled 'Ingresar Modalidad de Pago' is displayed. It contains the following fields and controls:

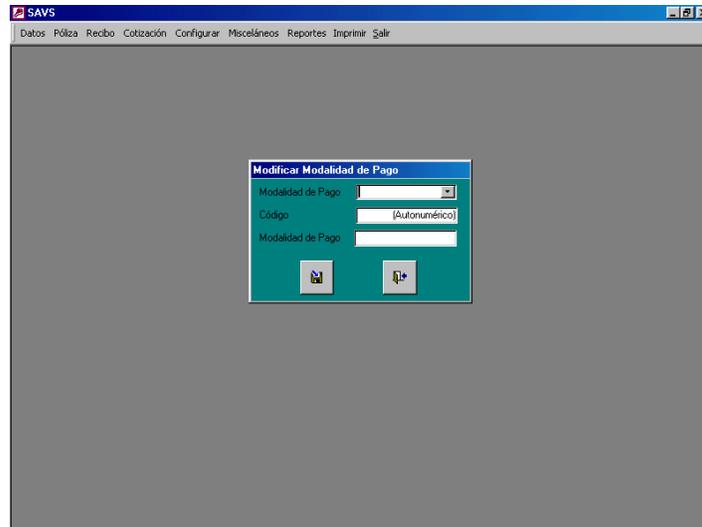
- 'Código': A text field with '(Autonumérico)' displayed inside.
- 'Modalidad de Pago': A text field.
- Two buttons at the bottom: a 'Guardar' (Save) button and a 'Salir' (Exit) button.

Esta opción se encuentra en **Configurar-Modalidad de Pago-Ingresar**.

Botón Guardar: Guarda la modalidad de pago siempre y cuando no se repita la modalidad.

Botón Salir: Cierra el formulario

Modificar Modalidad de Pago



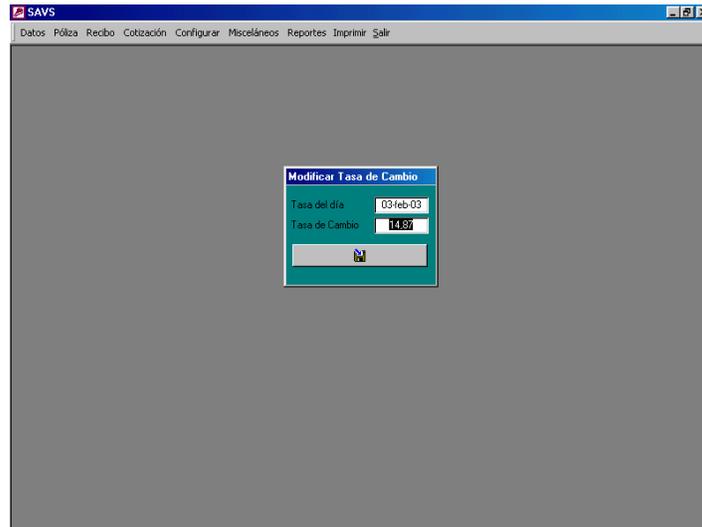
The image shows a screenshot of a software application window titled 'SAVS'. The menu bar includes 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. The main content area is a grey background with a central dialog box titled 'Modificar Modalidad de Pago'. This dialog box contains three input fields: 'Modalidad de Pago' (a dropdown menu), 'Código' (a text field with the value '[Autonumérico]'), and another 'Modalidad de Pago' (a text field). At the bottom of the dialog box are two buttons: a 'Guardar' button with a floppy disk icon and a 'Salir' button with a close icon.

Esta opción se encuentra en **Configurar-Modalidad de Pago-Modificar**. Se debe buscar la modalidad de pago a modificar en el campo *Modalidad de Pago*. El campo *código* no puede ser modificado.

Botón Guardar: Guarda los cambios realizados, siempre y cuando no se repita la modalidad.

Botón Salir: Cierra el formulario.

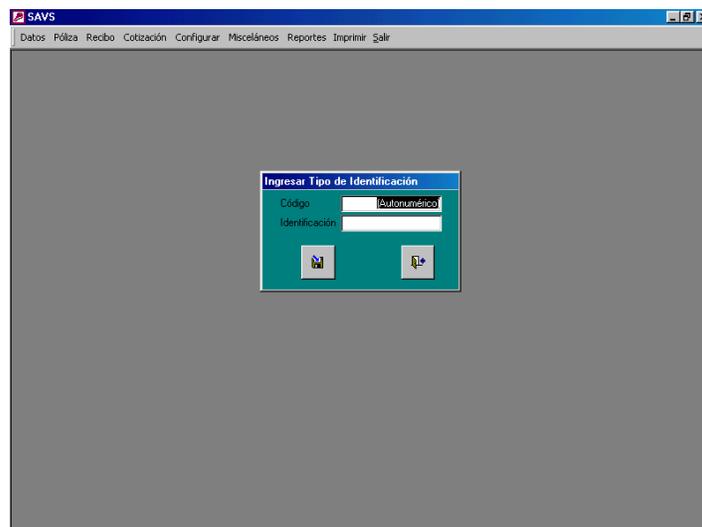
Modificar Tasa de Cambio



Esta opción se encuentra en **Configurar-Tasa de Cambio-Modificar**. Al abrir esta ventana se presentara la ultima tasa ingresada, es decir solamente se puede modificar la del día actual.

Botón Guardar: Guarda la tasa de cambio y cierra el formulario.

Ingresar Tipo de Identificación

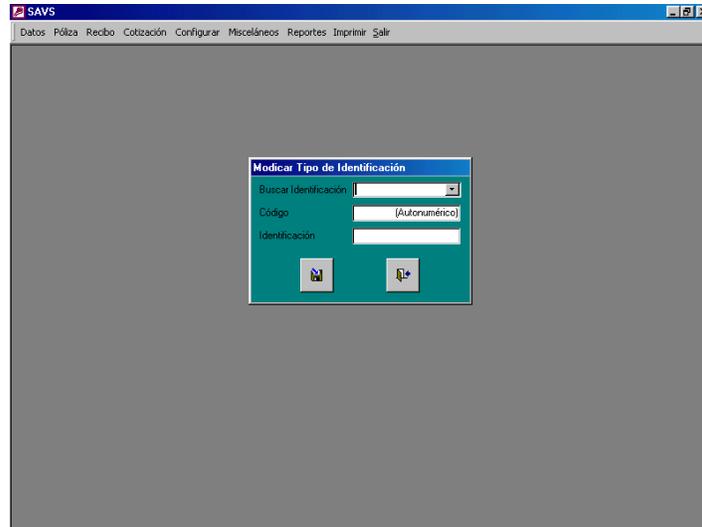


Esta opción se puede encontrar en **Configurar-Tipo de Identificación-Ingresar**.

Botón Guardar: Guarda el tipo de identificación siempre y cuando no se repita con uno antes guardado.

Botón Salir: Cierra el formulario

Modificar Tipo de Identificación

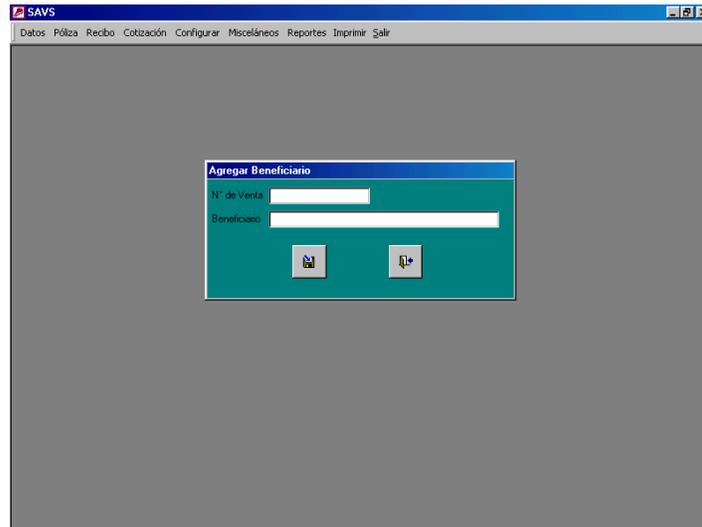


Esta opción se encuentra en **Configurar-Tipo de Identificación-Modificar**. Se debe buscar el tipo de identificación a modificar en el campo *Buscar Identificación*. El campo *código* no puede ser modificado.

Botón Guardar: Guarda los cambios realizados, siempre y cuando el tipo de identificación no se repita.

Botón Salir: Cierra el formulario.

Agregar Beneficiario a una Venta



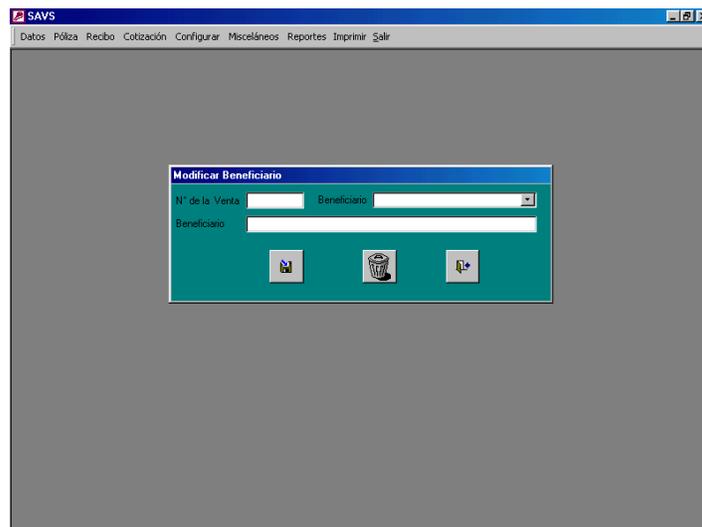
The screenshot shows a window titled 'SAVS' with a menu bar containing 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. In the center, there is a dialog box titled 'Agregar Beneficiario'. This dialog box has two text input fields: 'N° de Venta' and 'Beneficiario'. Below these fields are two buttons: a 'Guardar' button (represented by a floppy disk icon) and a 'Salir' button (represented by a window icon with a plus sign).

Esta opción esta disponible en **Misceláneos-Beneficiario-Agregar**. Para agregar un beneficiario a una venta, se debe ingresar el número de la venta en el campo *N de Venta*, después de verificar si existe el número y no esta en el limite de los permitidos se ingresa el beneficiario.

Botón Guardar: Guarda el beneficiario siempre y cuando no se repita con uno antes guardado.

Botón Salir: Cierra el formulario.

Modificar y Eliminar Beneficiario de una Venta



The screenshot shows a window titled 'SAVS' with the same menu bar as the previous image. In the center, there is a dialog box titled 'Modificar Beneficiario'. This dialog box has three input fields: 'N° de la Venta', 'Beneficiario' (which is a dropdown menu), and another 'Beneficiario' text input field. Below these fields are three buttons: a 'Guardar' button (floppy disk icon), a 'Eliminar' button (trash can icon), and a 'Salir' button (window icon with plus sign).

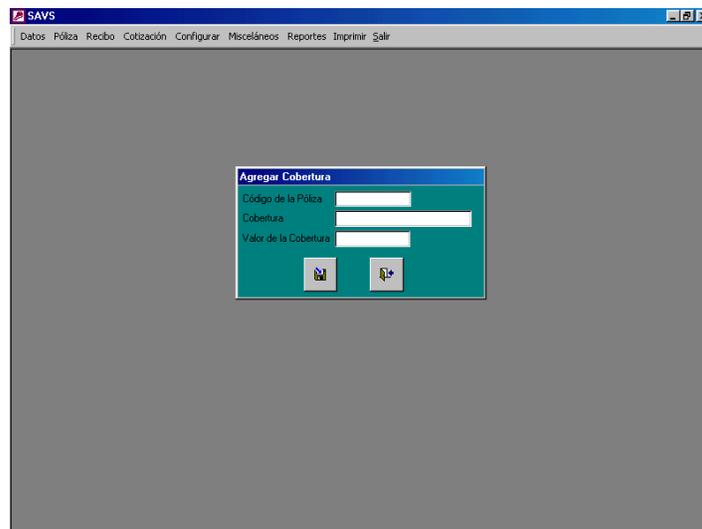
Esta opción esta disponible en **Misceláneos-Beneficiario-Modificar**. Para modificar un beneficiario de una venta, se debe ingresar el numero de la venta en el campo *N de Venta* y después buscar el beneficiario en el campo *Beneficiarios* que se desea modificar o eliminar.

Botón Guardar: Guarda el beneficiario siempre y cuando no se repita con uno antes guardado.

Botón Eliminar: Elimina el beneficiario seleccionado siempre y cuando no sea el único de esa venta.

Botón Salir: Cierra el formulario.

Agregar Cobertura a una Póliza



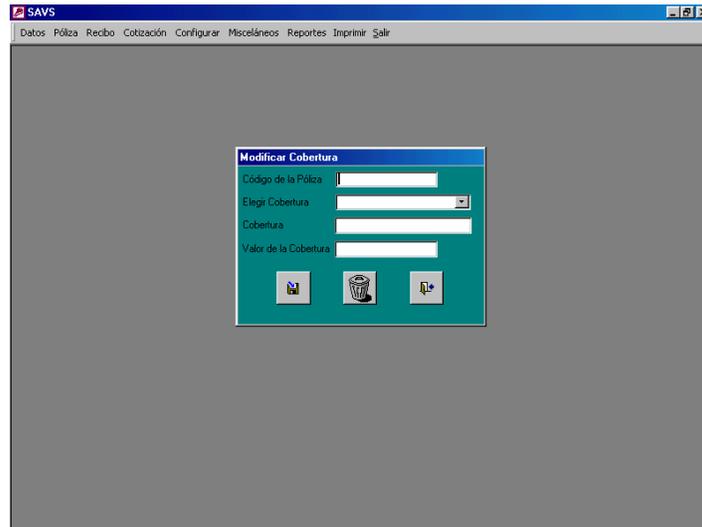
The image shows a screenshot of a web application window titled 'SAVS'. The main menu bar includes 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. The central content area is a grey rectangle containing a smaller form titled 'Agregar Cobertura'. This form has three input fields: 'Código de la Póliza', 'Cobertura', and 'Valor de la Cobertura'. Below the fields are two buttons: a blue button with a document icon and a grey button with a right-pointing arrow icon.

Esta opción esta disponible en **Misceláneos-Cobertura-Agregar**. Para agregar una cobertura a una póliza, se debe ingresar el código de la póliza en el campo *Código de la póliza*, después de verificar si existe el código se ingresa la cobertura.

Botón Guardar: Guarda la cobertura siempre y cuando no se repita con uno antes guardada.

Botón Salir: Cierra el formulario.

Modificar y Eliminar una Cobertura de una Póliza



The screenshot shows a window titled 'SAVS' with a menu bar containing 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. The main area is a grey background with a central dialog box titled 'Modificar Cobertura'. The dialog box has four input fields: 'Código de la Póliza' (text), 'Elegir Cobertura' (dropdown), 'Cobertura' (text), and 'Valor de la Cobertura' (text). Below the fields are three buttons: a floppy disk icon (Guardar), a trash can icon (Eliminar), and a document icon (Salir).

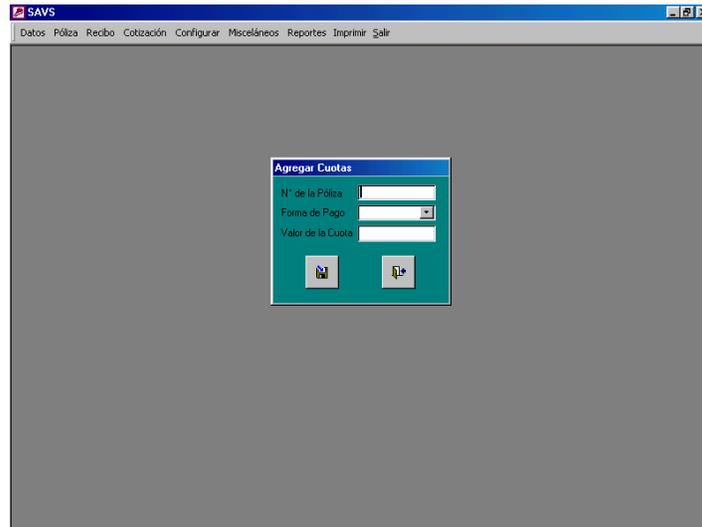
Esta opción está disponible en **Misceláneos-Cobertura-Modificar**. Para modificar una cobertura de una póliza, se debe ingresar el código de la póliza en el campo *Código de la Póliza* y después buscar la cobertura en el campo *Elegir cobertura* que se desea modificar o eliminar.

Botón Guardar: Guarda la cobertura siempre y cuando no se repita con una antes guardada.

Botón Eliminar: Elimina la cobertura seleccionada siempre y cuando no sea la única de esa póliza.

Botón Salir: Cierra el formulario.

Agregar Cuota a una Póliza



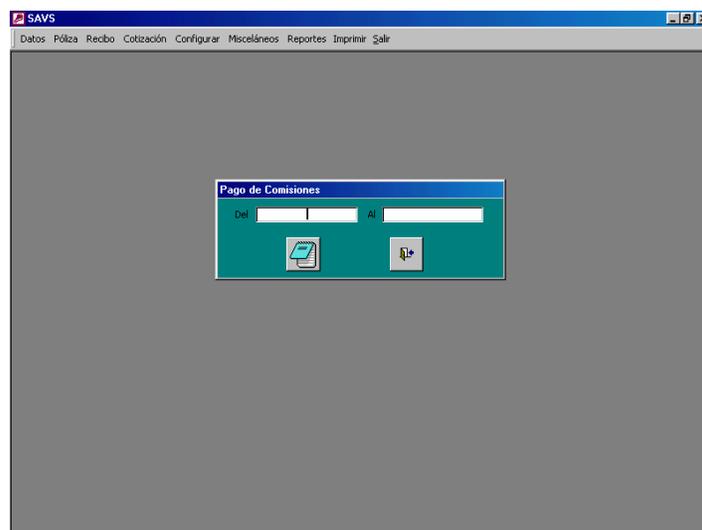
The screenshot shows a window titled 'SAVS' with a menu bar containing 'Datos', 'Póliza', 'Recibo', 'Cotización', 'Configurar', 'Misceláneos', 'Reportes', 'Imprimir', and 'Salir'. In the center, a dialog box titled 'Agregar Cuotas' is displayed. It contains three input fields: 'N° de la Póliza' (a text box), 'Forma de Pago' (a dropdown menu), and 'Valor de la Cuota' (a text box). Below these fields are two buttons: a 'Guardar' button with a floppy disk icon and a 'Salir' button with a door icon.

Esta opción está disponible en **Misceláneos-Cuota-Agregar**. Para agregar una cuota a una póliza, se debe ingresar el código de la póliza en el campo *N de la póliza*, después de verificar si existe el código se configura la nueva cuota.

Botón Guardar: Guarda la cuota siempre y cuando no se repita con una antes guardada.

Botón Salir: Cierra el formulario.

Generar Pago de Comisiones



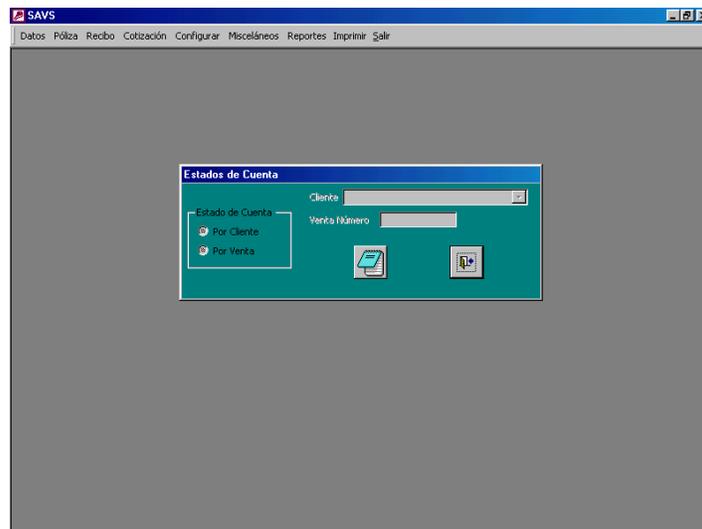
The screenshot shows the same 'SAVS' window. In the center, a dialog box titled 'Pago de Comisiones' is displayed. It features two text input fields labeled 'Del' and 'Al'. Below these fields are two buttons: a 'Generar' button with a document icon and a 'Salir' button with a door icon.

Esta opción se encuentra en **Reportes-Comisiones**. El pago de comisiones se genera en un intervalo de tiempo el cual es ingresado en los campos *Del* y *Al*.

Botón Generar Comisiones: Ejecuta el reportes de comisiones.

Botón Salir: Cierra el formulario.

Generar Estado de Cuenta



Esta opción se encuentra en **Reportes-Estados de Cuenta**. Los reportes de Estado de Cuenta se pueden generar por Cliente o por Venta.

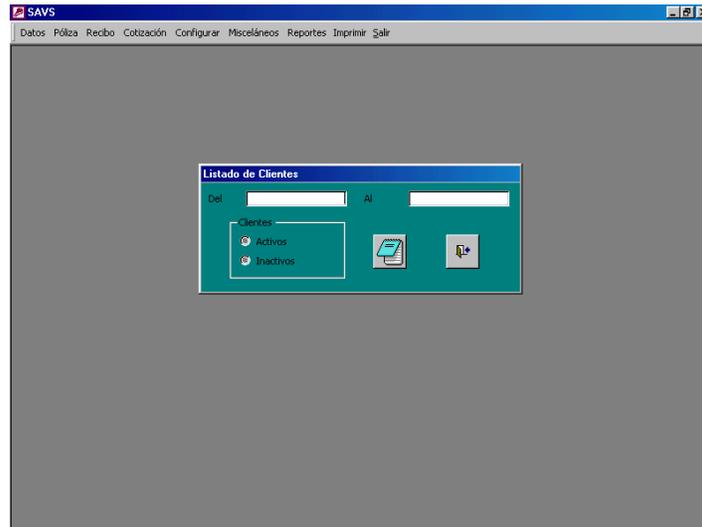
Cuando se genera por Cliente, se debe seleccionar el nombre en el campo *Cliente*, si ese cliente tiene asociada mas de una venta se generara un estado de cuenta para cada una.

Cuando se genera por Venta, se debe ingresar el numero de la venta en el campo *Venta Numero*, generándose el estado de cuenta para esa venta.

Botón Generar Estados de Cuenta: Ejecuta el reporte según los parámetros.

Botón Salir: Cierra el formulario.

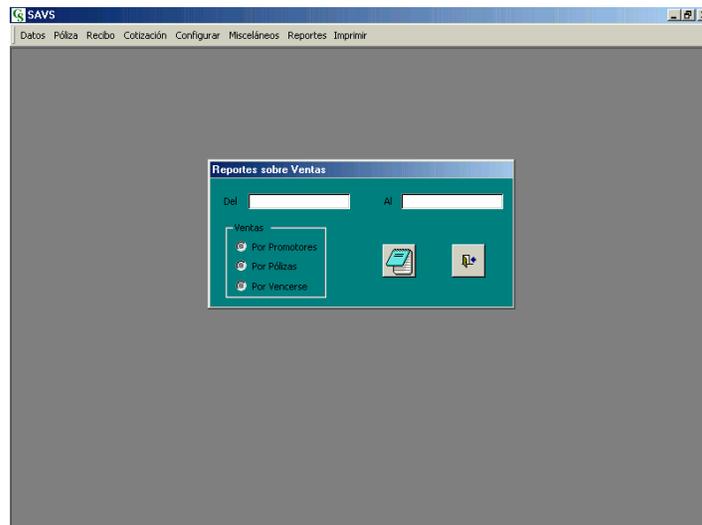
Generar Listado de Clientes



Esta opción se encuentra en **Reportes-Listado de Clientes**. El listado se puede realizar por clientes activos y no activos en una fecha determinada.

Un Cliente es considerado inactivo cuando no tiene ninguna venta vigente asociada.

Generar Reportes de Ventas



Esta opción se encuentra en **Reportes-Reportes de Ventas**. El reporte sobre venta se puede realizar global, por pólizas, por promotores y por vencerse en una fecha determinada.

El reporte global se ejecuta cuando no se escoge ninguna de las tres opciones y toma como parámetro el intervalo de fecha.

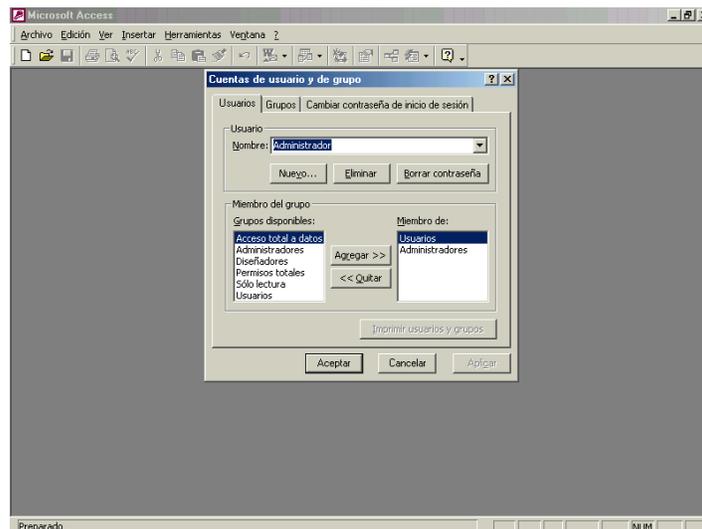
El reporte ventar por vencerse selecciona las ventas que van a caducar en un intervalo de tiempo.

Botón Generar: Ejecuta el reporte según los parámetros.

Botón Salir: Cierra el formulario.

Crear y Eliminar Usuarios

Para crear un nuevo usuario en el sistema se requiere que el usuario a realizar la acción posea los permisos de administrador. Al presentarse la ventana de inicio de sesión, se debe cancelar, luego se debe acceder al menú de **Herramientas-Seguridad-Cuentas de usuario y de grupo...**, donde luego le aparecerá nuevamente la ventana de inicio de sesión donde debe ingresar.



Cambiar Contraseña

Para cambiar su contraseña en el sistema, se requiere que el usuario a realizar la acción cancele el inicio de sesión, luego debe acceder al menú de **Herramientas-Seguridad-Cuentas de usuario y de grupo...**, donde luego le aparecerá nuevamente la ventana de inicio de sesión donde debe ingresar y cambiar su contraseña.

CODIGO FUENTE

Ver archivo [Codigo Fuente.pdf](#)

Form_AGREGAR_BENEFICIARIO - 1

```
Option Compare Database
Public POSINI As Integer
Private Sub BUSCAR_AfterUpdate()
Dim rs As Object
Set rs = Me.Recordset.Clone
rs.FindFirst "[ID_VENTA] = " & Str(Me![BUSCAR])
Me.Bookmark = rs.Bookmark
DoCmd.Requery "CUENTA"
If ID_VENTA = BUSCAR Then
    If CUENTA.ListCount < CANT_BENEFIC Then
        BENEFICIARIO.Locked = False
        DoCmd.GoToControl "BENEFICIARIO"
        BUSCAR.Locked = True
        POSINI = Form_AGREGAR_BENEFICIARIO.CurrentRecord
        DoCmd.GoToRecord , , acNewRec
    Else
        Inicio
        MsgBox "Esta Venta tiene el límite de Beneficiarios permitidos", vbInformation, "SAVS"
    End If
Else
    Inicio
    MsgBox "Esta Venta no existe", vbInformation, "SAVS"
End If

End Sub

Private Sub BUSCAR_KeyPress(Keycode As Integer)
If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 45
Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub Form_Open(Cancel As Integer)
ValSalir
Inicio
End Sub
Sub Inicio()
DoCmd.GoToRecord , , acNewRec
CANT_BENEFIC.Visible = False
ID_POL.Visible = False
ID_VENTA.Visible = False
CUENTA.Visible = False
BENEFICIARIO.Locked = True
BUSCAR.Locked = False
BUSCAR = ""
DoCmd.GoToControl "BUSCAR"
End Sub
Private Sub btnGuardar_Click()
If BUSCAR <> "" Then
var = BENEFICIARIO
BAN = 0
pos = Form_AGREGAR_BENEFICIARIO.CurrentRecord

If BENEFICIARIO <> "" Then
    ID_VENTA = BUSCAR
    DoCmd.GoToRecord , , acGoTo, POSINI
```

```
For I = 1 To CUENTA.ListCount - 1
  If var = BENEFICIARIO Then
    MsgBox "Beneficiario existente", vbInformation, "SAVS"
    DoCmd.GoToRecord , , acGoTo, pos
    DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
    DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
    I = CUENTA.ListCount + 1
    BAN = 1
  End If
  If BAN <> 1 And Form_AGREGAR_BENEFICIARIO.CurrentRecord <> POSINI + CUENTA.ListCount - 2
Then
    DoCmd.GoToRecord , , acNext
  End If
Next I
Inicio
If BAN = 0 Then
  MsgBox "Beneficiario Guardado", vbInformation, "SAVS"
End If
Else
  MsgBox "Ingrese el nombre del Beneficiario", vbInformation, "SAVS"
End If
Else
  MsgBox "Ingrese el número de la Venta", vbInformation, "SAVS"
End If

End Sub

Private Sub btnSalir_Click()
If BENEFICIARIO.Locked = True Or IsNull(BENEFICIARIO) = True Then
  DoCmd.Close
  ValSalir
Else
  MsgBox "Debe guardar ante de salir", vbInformation, "SAVS"
End If

End Sub
```

Form_AGREGAR_COBERTURA - 1

```
Option Compare Database
Public POSINI As Integer
Sub Inicio()
DoCmd.GoToRecord , , acNewRec
BUSCAR.Locked = False
ID_POL.Visible = False
BUSCAR_COBERTURA.Visible = False
COBERTURA.Locked = True
VALOR_COBER.Locked = True
BUSCAR = ""
BUSCAR_COBERTURA = ""
End Sub
Private Sub BUSCAR_AfterUpdate()
' Buscar el registro que coincida con el control.
Dim rs As Object

Set rs = Me.Recordset.Clone
rs.FindFirst "[ID_POL] = '" & Me![BUSCAR] & "'"
Me.Bookmark = rs.Bookmark
DoCmd.Requery "BUSCAR_COBERTURA"
If ID_POL = BUSCAR Then
    COBERTURA.Locked = False
    VALOR_COBER.Locked = False
    POSINI = Form_AGREGAR_COBERTURA.CurrentRecord
    DoCmd.GoToRecord , , acNewRec
    DoCmd.GoToControl "COBERTURA"
    BUSCAR.Locked = True
Else
    MsgBox "Esta Póliza no existe", vbInformation, "SAVS"
    Inicio
End If
End Sub

Private Sub Form_Open(Cancel As Integer)
ValSalir
Inicio
End Sub

Private Sub VALOR_COBER_KeyPress(Keycode As Integer)
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 45 Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub
Private Sub btnGuardar_Click()
If BUSCAR <> "" Then
var = COBERTURA
BAN = 0
pos = Form_AGREGAR_COBERTURA.CurrentRecord

If COBERTURA <> "" And VALOR_COBER <> 0 Then
    ID_POL = BUSCAR
    DoCmd.GoToRecord , , acGoTo, POSINI
    For I = 1 To BUSCAR_COBERTURA.ListCount - 1
        If var = COBERTURA Then
            MsgBox "Cobertura existente", vbInformation, "SAVS"
```

```
        DoCmd.GoToRecord , , acGoTo, pos
        DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
        DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
        I = BUSCAR_COBERTURA.ListCount + 1
        BAN = 1
    End If
    If BAN <> 1 And Form_AGREGAR_COBERTURA.CurrentRecord <> POSINI + BUSCAR_COBERTURA.ListCo
unt - 2 Then
        DoCmd.GoToRecord , , acNext
    End If
Next I
Inicio
If BAN = 0 Then
    MsgBox "Cobertura Guardada", vbInformation, "SAVS"
End If
Else
    MsgBox "Ingrese los datos de la Cobertura", vbInformation, "SAVS"
End If
Else
    MsgBox "Ingrese el Código de la Póliza", vbInformation, "SAVS"
End If
End Sub
Private Sub btnSalir_Click()
If COBERTURA.Locked = True Or (IsNull(COBERTURA) = True And IsNull(VALOR_COBER) = True) Then
    DoCmd.Close
    ValSalir
Else
    MsgBox "Debe guardar antes de Salir", vbInformation, "SAVS"
End If
End Sub
```

Form_AGREGAR_CUOTA - 1

```
Option Compare Database
Public POSINI As Integer
Sub Inicio()
DoCmd.GoToRecord , , acNewRec
BUSCAR = ""
BUSCAR.Locked = False
ID_POL.Visible = False
CANT_PAGOS.Visible = False
VALOR_VENTA.Visible = False
BUSCAR_CUOTA.Visible = False
FORMA_PAGO.Locked = True
VALOR_CUOTA.Locked = True

End Sub

Private Sub BUSCAR_AfterUpdate()
' Buscar el registro que coincida con el control.
Dim rs As Object

Set rs = Me.Recordset.Clone
rs.FindFirst "[ID_POL] = '" & Me![BUSCAR] & "'"
Me.Bookmark = rs.Bookmark
DoCmd.Requery "BUSCAR_CUOTA"
If ID_POL = BUSCAR Then
FORMA_PAGO.Locked = False
DoCmd.GoToControl "FORMA_PAGO"
POSINI = Form_AGREGAR_CUOTA.CurrentRecord
DoCmd.GoToRecord , , acNewRec
BUSCAR.Locked = True

Else
MsgBox "Esta Póliza no existe", vbInformation, "SAVS"
Inicio
End If
End Sub

Private Sub Form_Open(Cancel As Integer)
ValSalir
Inicio
End Sub

Private Sub btnGuardar_Click()
If BUSCAR <> "" Then
BAN = 0
pos = Form_AGREGAR_CUOTA.CurrentRecord
If FORMA_PAGO <> 0 Then
var = FORMA_PAGO
ID_POL = BUSCAR
DoCmd.GoToRecord , , acGoTo, POSINI
For I = 1 To BUSCAR_CUOTA.ListCount - 1
If var = FORMA_PAGO Then
MsgBox "Forma de Pago existente", vbInformation, "SAVS"
DoCmd.GoToRecord , , acGoTo, pos
DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
I = BUSCAR_CUOTA.ListCount + 1
BAN = 1
End If
If BAN <> 1 And Form_AGREGAR_CUOTA.CurrentRecord <> POSINI + BUSCAR_CUOTA.ListCount - 2
Then
DoCmd.GoToRecord , , acNext
End If
Next I
```

Form_AGREGAR_CUOTA - 2

```
Inicio
If BAN = 0 Then
    MsgBox "Cuota Guardada", vbInformation, "SAVS"
End If
Else
    MsgBox "Ingrese los datos de la Cuota", vbInformation, "SAVS"
End If
Else
    MsgBox "Ingrese el Código de la Póliza", vbInformation, "SAVS"
End If

End Sub

Private Sub btnSalir_Click()
If IsNull(FORMA_PAGO) = True Then
    DoCmd.Close
    ValSalir
Else
    MsgBox "Debe guardar antes de Salir", vbInformation, "SAVS"
End If
End Sub

Private Sub FORMA_PAGO_AfterUpdate()
ID_POL = Form_POLIZA_COMPRA.ID_POL
VALOR_CUOTA = VALOR_VENTA / CANT_PAGOS
End Sub

Private Sub FORMA_PAGO_KeyPress(Keycode As Integer)
If Keycode >= 0 And Keycode <= 255 Then
    Keycode = 0
    Beep
End If

End Sub
```

Form_BANCOS - 1

Option Compare Database

```
Private Sub btnGuardar_Click()  
BAN = 0  
DoCmd.GoToRecord , , acLast  
If BANCO <> "" Then  
    var = BANCO  
    HASTA = Form_BANCOS.CurrentRecord  
    If HASTA = 1 Then  
        DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70  
        DoCmd.GoToRecord , , acNewRec  
    Else  
        For I = HASTA - 1 To 1 Step -1  
            DoCmd.GoToRecord , , acPrevious  
            If var = BANCO Then  
                MsgBox "Banco existente", vbInformation, "SAVS"  
                DoCmd.GoToRecord , , acLast  
                I = 0  
                BAN = 1  
            End If  
        Next I  
        If BAN = 0 Then  
            DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70  
            DoCmd.GoToRecord , , acNewRec  
        End If  
    End If  
Else  
    MsgBox "Ingrese el nombre del Banco", vbInformation, "SAVS"  
End If  
  
End Sub  
Private Sub btnSalir_Click()  
  
If ID_BANCO <> "" Then  
    If BANCO <> "" Then  
        MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"  
    Else  
        DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70  
        DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70  
        DoCmd.Close  
        ValSalir  
    End If  
Else  
    DoCmd.Close  
    ValSalir  
End If  
  
End Sub  
  
Private Sub Form_Open(Cancel As Integer)  
ValSalir  
DoCmd.GoToRecord , , acNewRec  
End Sub
```

Form_BENEFICIARIOS - 1

Option Compare Database

```
Private Sub Form_Open(Cancel As Integer)
ValSalir
DoCmd.GoToRecord , , acNewRec
ID_VENTA.Locked = True
ID_POL.Locked = True
CANT_BENEFIC.Locked = True
End Sub

Private Sub btnGuardar_Click()
If BENEFICIARIO.Locked = False Then
BAN = 0
DoCmd.GoToRecord , , acLast
If BENEFICIARIO <> "" Then
var = BENEFICIARIO
HASTA = Form_BENEFICIARIOS.CurrentRecord
If HASTA = 1 Then
Guardar
Else
For I = HASTA - 1 To 1 Step -1
DoCmd.GoToRecord , , acPrevious
If var = BENEFICIARIO Then
MsgBox "Beneficiario existente", vbInformation, "SAVS"
DoCmd.GoToRecord , , acLast
I = 0
BAN = 1
End If
Next I
If BAN = 0 Then
Guardar
End If
End If
Else
MsgBox "Ingrese el nombre del Beneficiario", vbInformation, "SAVS"
End If
Else
MsgBox "Este es el límite de Beneficiarios", vbInformation, "SAVS"
End If
End Sub

Sub Guardar()
DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
DoCmd.GoToRecord , , acNewRec
Form_BENEFICIARIOS.ID_VENTA = Form_VENTAS.ID_VENTA
If CANT_BENEFIC < Form_BENEFICIARIOS.CurrentRecord Then
MsgBox "Este es el límite de Beneficiarios", vbInformation, "SAVS"
BENEFICIARIO.Locked = True
End If

End Sub

Private Sub btnSalir_Click()
If Form_BENEFICIARIOS.CurrentRecord > 1 Then
If BENEFICIARIO <> "" Then
MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"
Else
Cerrar_Abrir
End If
Else
MsgBox "Toda Venta debe tener al menos un Beneficiario", vbInformation, "SAVS"
End If
```

End Sub

Sub Cerrar_Abrir()

Dim stLinkCriteria As String

stLinkCriteria = "[ID_VENTA]=" & Me![ID_VENTA]

DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70

DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70

DoCmd.Close

DoCmd.OpenForm "CANCEL_RECIBO1", , , stLinkCriteria

End Sub

Option Compare Database

```
Private Sub BUSCAR_AfterUpdate()  
Dim rs As Object  
Set rs = Me.Recordset.Clone  
rs.FindFirst "[ID_CLIENTE] = " & Str(Me![BUSCAR])  
Me.Bookmark = rs.Bookmark  
If ID_CLIENTE = BUSCAR Then  
  
Else  
    DoCmd.GoToRecord , , acNewRec  
    MsgBox "Este Cliente no existe", vbInformation, "SAVS"  
    BUSCAR = ""  
    DoCmd.GoToControl "BUSCAR"  
End If  
  
End Sub  
  
Private Sub BUSCAR_KeyPress(Keycode As Integer)  
If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13  
Or Keycode = 28 Or Keycode = 9 Then  
Else  
    Keycode = 0  
    Beep  
End If  
  
End Sub  
  
Private Sub Form_Open(Cancel As Integer)  
DoCmd.GoToRecord , , acNewRec  
DoCmd.GoToControl "BUSCAR"  
ID_CLIENTE.Visible = False  
FECHA_VISITA.Locked = True  
NOMBRE.Locked = True  
DIRECCION.Locked = True  
TELEFONO.Locked = True  
CELULAR.Locked = True  
TIPO_ID.Locked = True  
NUM_ID.Locked = True  
End Sub  
  
Private Sub btnSalir_Click()  
DoCmd.Close  
End Sub
```

Form_BUSCAR_VENTAS - 1

Option Compare Database

Public var As Integer

Private Sub Form_Activate()

var = Form_BUSCAR_VENTAS.CurrentRecord

If var = 0 Then

MsgBox "Esta Venta no existe", vbInformation, "SAVS"

Else

ID_VENTA.Visible = True

FECHA_ADQUI.Locked = True

CLIENTES_NOMBRE.Locked = True

POLIZA_COMPRA_DESCRIPCION.Locked = True

PROMOTORES_NOMBRE.Locked = True

DESCRIPCION.Locked = True

End If

End Sub

Private Sub btnSalir_Click()

DoCmd.Close acForm, "PARAMETRO_VENTAS"

DoCmd.Close

End Sub

Private Sub Form_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)

If var = 0 Then

DoCmd.Close

End If

End Sub

Option Compare Database

Sub Inicio()

TASA_CAMBIO.Visible = False

DoCmd.GoToControl "BUSCAR"

ID_RECIBO.Visible = False

FECHA_CANCEL.Visible = False

ESTADO.Visible = False

VALOR_TOTAL_CORDOBAS.Visible = False

CORDOBAS.Locked = True

DOLARES.Locked = True

TASA.Locked = True

NOMBRE.Locked = True

VALOR_RECIBO.Locked = True

TOTAL_CORDOBAS.Locked = True

LETRAS.Locked = True

TIPO_PAGO.Locked = True

ID_BANCO.Locked = True

REFERENCIA.Locked = True

BUSCAR.Locked = False

TOTAL_CORDOBAS = ""

TASA = ""

DoCmd.GoToRecord , , acNewRec

End Sub

Sub Cancelar()

ESTADO = "CANCELADO"

FECHA_CANCEL = Date

MsgBox "Recibo Cancelado N° " & ID_RECIBO, vbInformation, "SAVS"

DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70

End Sub

Private Sub btnCancelar_Click()

If DOLARES <> "" Then

Else

DOLARES = 0

End If

If CORDOBAS <> "" Then

Else

CORDOBAS = 0

End If

monto = VALOR_RECIBO * TASA

VALOR_TOTAL_CORDOBAS = (DOLARES * TASA) + CORDOBAS

If TIPO_PAGO <> 0 Then

If TIPO_PAGO <> 1 Then

If REFERENCIA <> "" And ID_BANCO <> 0 Then

If monto = VALOR_TOTAL_CORDOBAS Then

If LETRAS <> "" Then

Cancelar

Inicio

DoCmd.OpenReport "IMPRIMIR_RECIBO_NOEFECTIVO", acViewPreview

BUSCAR = ""

Else

MsgBox "Introduzca el Valor en Letras", vbInformation, "SAVS"

End If

Else

MsgBox "El monto recibido no equivale al valor de la cuota", vbInformation, "SAVS"

S"

End If

Else

MsgBox "Ingrese los datos de la Modalidad de Pago", vbInformation, "SAVS"

```

        End If
    Else
        If monto = VALOR_TOTAL_CORDOBAS Then
            If LETRAS <> "" Then
                Cancelar
                Inicio
                DoCmd.OpenReport "IMPRIMIR_RECIBO_EFECTIVO", acViewPreview
                BUSCAR = ""
            Else
                MsgBox "Introduzca el Valor en Letras", vbInformation, "SAVS"
            End If
        Else
            MsgBox "El monto recibido no equivale al valor de la cuota", vbInformation, "SAVS"
        End If
    End If

Else
    MsgBox "Elija la Modalidad de Pago", vbInformation, "SAVS"
End If

End Sub

Private Sub BUSCAR_AfterUpdate()
    Dim rs As Object
    Set rs = Me.Recordset.Clone
    rs.FindFirst "[ID_RECIBO] = " & Str(Me![BUSCAR])
    Me.Bookmark = rs.Bookmark

    If ID_RECIBO = BUSCAR Then
        If ESTADO = "EMITIDO" Then
            BUSCAR.Locked = True
            TIPO_PAGO.Locked = False
            CORDOBAS.Locked = False
            DOLARES.Locked = False
            TASA = Forms!CANCEL_RECIBO![TASA_CAMBIO].Form![TASA]
            TOTAL_CORDOBAS = VALOR_RECIBO * TASA
            DoCmd.RunMacro "COPIA_PEGA"
            SendKeys "{TAB}"
        Else
            Inicio
            MsgBox "Está Cancelado el Recibo N° " & BUSCAR, vbInformation, "SAVS"
            BUSCAR = ""
            DoCmd.GoToControl "BUSCAR"
        End If
    Else
        Inicio
        MsgBox "No existe el Recibo N° " & BUSCAR, vbInformation, "SAVS"
        BUSCAR = ""
        DoCmd.GoToControl "BUSCAR"
    End If

End Sub

Private Sub BUSCAR_KeyPress(Keycode As Integer)
    If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
    Else
        Keycode = 0
        Beep
    End If

End Sub

```

```
Private Sub CORDOBAS_KeyPress(Keycode As Integer)
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub
```

```
Private Sub DOLARES_KeyPress(Keycode As Integer)
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub
```

```
Private Sub Form_Open(Cancel As Integer)
Inicio
End Sub
```

```
Private Sub ID_BANCO_KeyPress(Keycode As Integer)
If Keycode >= 0 And Keycode <= 255 Then
    Keycode = 0
    Beep
End If

End Sub
```

```
Private Sub NUMERO_LostFocus()
CENTAVOS = "CORDOBAS CON " & Right(NUMERO, 2) & "/100 CENTAVOS"
Select Case Len(NUMERO)
Case 4
    LETRAS = UNIDAD(NUMERO) & " " & CENTAVOS
Case 5
    LETRAS = DECENA(NUMERO) & " " & CENTAVOS
Case 6
    LETRAS = CENTENA(NUMERO) & " " & CENTAVOS
Case 8
    LETRAS = MILLAR1(NUMERO) & " " & CENTAVOS
Case 9
    LETRAS = MILLAR2(NUMERO) & " " & CENTAVOS
Case 10
    LETRAS = MILLAR3(NUMERO) & " " & CENTAVOS
Case 12
    LETRAS = MILLON1(NUMERO) & " " & CENTAVOS
End Select

End Sub
```

```
Private Sub TASA_KeyPress(Keycode As Integer)
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 45 Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If
```

End Sub

```
Private Sub TIPO_PAGO_AfterUpdate()
```

```
If TIPO_PAGO <> 1 Then
```

```
    REFERENCIA.Locked = False
```

```
    ID_BANCO.Locked = False
```

```
Else
```

```
    REFERENCIA = ""
```

```
    ID_BANCO = ""
```

```
    REFERENCIA.Locked = True
```

```
    ID_BANCO.Locked = True
```

```
End If
```

End Sub

```
Private Sub TIPO_PAGO_KeyPress(Keycode As Integer)
```

```
If Keycode >= 0 And Keycode <= 255 Then
```

```
    Keycode = 0
```

```
    Beep
```

```
End If
```

End Sub

```
Private Sub btnSalir_Click()
```

```
    respuesta = MsgBox("¿Desea Salir sin Cancelar Recibo?", vbYesNo, "SAVS")
```

```
If respuesta = vbYes Then
```

```
    DOLARES = ""
```

```
    CORDOBAS = ""
```

```
    LETRAS = ""
```

```
    REFERENCIA = ""
```

```
    ID_BANCO = ""
```

```
    TIPO_PAGO = ""
```

```
    VALOR_TOTAL_CORDOBAS = ""
```

```
    DoCmd.Close
```

```
Else
```

```
End If
```

End Sub

```
'****FUNCIONES PARA ESCRIBIR EL VALOR DEL RECIBO****
```

```
Function UNIDAD(NUMERO) As String
```

```
Select Case Left(Right(NUMERO, 4), 1)
```

```
Case 1
```

```
    UNIDAD = "UN"
```

```
Case 2
```

```
    UNIDAD = "DOS"
```

```
Case 3
```

```
    UNIDAD = "TRES"
```

```
Case 4
```

```
    UNIDAD = "CUATRO"
```

```
Case 5
```

```
    UNIDAD = "CINCO"
```

```
Case 6
```

```
    UNIDAD = "SEIS"
```

```
Case 7
```

```
    UNIDAD = "SIETE"
```

```
Case 8
```

```
    UNIDAD = "OCHO"
```

```
Case 9
```

```
    UNIDAD = "NUEVE"
```

```
End Select
```

```
End Function
```

```
Function DECENA(NUMERO) As String
```

```
Select Case Left(Right(NUMERO, 5), 2)
```

```

Case 10
    DECENA = "DIEZ "
Case 11
    DECENA = "ONCE "
Case 12
    DECENA = "DOCE "
Case 13
    DECENA = "TRECE "
Case 14
    DECENA = "CATORCE "
Case 15
    DECENA = "QUINCE "
Case 16 To 19
    DECENA = "DIEZ Y " & UNIDAD(Right(NUMERO, 4))
Case 20
    DECENA = "VEIENTE "
Case 21 To 29
    DECENA = "VEINTE Y " & UNIDAD(Right(NUMERO, 4))
Case 30
    DECENA = "TREINTA " & UNIDAD(Right(NUMERO, 4))
Case 31 To 39
    DECENA = "TREINTA Y " & UNIDAD(Right(NUMERO, 4))
Case 40
    DECENA = "CUARENTA "
Case 41 To 49
    DECENA = "CUARENTA Y " & UNIDAD(Right(NUMERO, 4))
Case 50
    DECENA = "CINCUENTA "
Case 51 To 59
    DECENA = "CINCUENTA Y " & UNIDAD(Right(NUMERO, 4))
Case 60
    DECENA = "SESENTA "
Case 61 To 69
    DECENA = "SESENTA Y " & UNIDAD(Right(NUMERO, 4))
Case 70
    DECENA = "SETENTA"
Case 71 To 79
    DECENA = "SETENTA Y " & UNIDAD(Right(NUMERO, 4))
Case 80
    DECENA = "OCHENTA"
Case 81 To 89
    DECENA = "OCHENTA Y " & UNIDAD(Right(NUMERO, 4))
Case 90
    DECENA = "NOVENTA"
Case 91 To 99
    DECENA = "NOVENTA Y " & UNIDAD(Right(NUMERO, 4))
End Select
End Function

```

```

Function CENTENA(NUMERO) As String
Select Case Left(Right(NUMERO, 6), 3)
Case 100
    CENTENA = "CIEN "
Case 101 To 109
    CENTENA = "CIENTO " & UNIDAD(Right(NUMERO, 4))
Case 110 To 199
    CENTENA = "CIENTO " & DECENA(Right(NUMERO, 5))
Case 200, 300, 400, 600, 800
    CENTENA = UNIDAD(Left(NUMERO, 4)) & "CIENTOS "
Case 201 To 209
    CENTENA = "DOSCIENTOS " & UNIDAD(Right(NUMERO, 4))

```

```

Case 210 To 299
    CENTENA = "DOSCIENTOS " & DECENA(Right(NUMERO, 5))
Case 301 To 309
    CENTENA = "TRESCIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 310 To 399
    CENTENA = "TRESCIENTOS " & DECENA(Right(NUMERO, 5))
Case 401 To 409
    CENTENA = "CUATROCIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 410 To 499
    CENTENA = "CUATROCIENTOS " & DECENA(Right(NUMERO, 5))
Case 500
    CENTENA = "QUINIENTOS "
Case 501 To 509
    CENTENA = "QUINIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 510 To 599
    CENTENA = "QUINIENTOS " & DECENA(Right(NUMERO, 5))
Case 601 To 609
    CENTENA = "SEISCIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 610 To 699
    CENTENA = "SEISCIENTOS " & DECENA(Right(NUMERO, 5))
Case 700
    CENTENA = "SETECIENTOS "
Case 701 To 709
    CENTENA = "SETECIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 710 To 799
    CENTENA = "SETECIENTOS " & DECENA(Right(NUMERO, 5))
Case 801 To 809
    CENTENA = "OCHOCIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 810 To 899
    CENTENA = "OCHOCIENTOS " & DECENA(Right(NUMERO, 5))
Case 900
    CENTENA = "NOVECIENTOS "
Case 901 To 909
    CENTENA = "NOVECIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 910 To 999
    CENTENA = "NOVECIENTOS " & DECENA(Right(NUMERO, 5))
End Select
End Function

```

```

Function MILLAR1(NUMERO) As String
If Mid(NUMERO, 3, 1) = 0 Then
    If Mid(NUMERO, 4, 1) = 0 Then
        If Mid(NUMERO, 5, 1) = 0 Then
            MILLAR1 = UNIDAD(Left(NUMERO, 4)) & " MIL"
        Else
            MILLAR1 = UNIDAD(Left(NUMERO, 4)) & " MIL " & UNIDAD(Right(NUMERO, 4))
        End If
    Else
        MILLAR1 = UNIDAD(Left(NUMERO, 4)) & " MIL " & DECENA(Right(NUMERO, 5))
    End If
Else
MILLAR1 = UNIDAD(Left(NUMERO, 4)) & " MIL " & CENTENA(Right(NUMERO, 6))
End If

```

```

End Function
Function MILLAR2(NUMERO) As String
If Mid(NUMERO, 4, 1) = 0 Then
    If Mid(NUMERO, 5, 1) = 0 Then
        If Mid(NUMERO, 6, 1) = 0 Then
            MILLAR2 = DECENA(Left(NUMERO, 5)) & " MIL"
        Else

```

```

        MILLAR2 = DECENA(Left(NUMERO, 5)) & " MIL " & UNIDAD(Right(NUMERO, 4))
    End If
Else
    MILLAR2 = DECENA(Left(NUMERO, 5)) & " MIL " & DECENA(Right(NUMERO, 5))
End If
Else
    MILLAR2 = DECENA(Left(NUMERO, 5)) & " MIL " & CENTENA(Right(NUMERO, 6))
End If
End Function
Function MILLAR3(NUMERO) As String
If Mid(NUMERO, 5, 1) = 0 Then
    If Mid(NUMERO, 6, 1) = 0 Then
        If Mid(NUMERO, 7, 1) = 0 Then
            MILLAR3 = CENTENA(Left(NUMERO, 6)) & " MIL"
        Else
            MILLAR3 = CENTENA(Left(NUMERO, 6)) & " MIL " & UNIDAD(Right(NUMERO, 4))
        End If
    Else
        MILLAR3 = CENTENA(Left(NUMERO, 6)) & " MIL " & DECENA(Right(NUMERO, 5))
    End If
Else
    MILLAR3 = CENTENA(Left(NUMERO, 6)) & " MIL " & CENTENA(Right(NUMERO, 6))
End If
End Function
Function MILLON1(NUMERO) As String
If Mid(NUMERO, 1, 1) = 1 Then
If Mid(NUMERO, 3, 1) = 0 Then
    If Mid(NUMERO, 4, 1) = 0 Then
        If Mid(NUMERO, 5, 1) = 0 Then
            If Mid(NUMERO, 7, 1) = 0 Then
                If Mid(NUMERO, 8, 1) = 0 Then
                    If Mid(NUMERO, 9, 1) = 0 Then
                        MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON DE "
                    Else
                        MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON " & UNIDAD(Right(NUMERO, 4))
                    )
                End If
            Else
                MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON " & DECENA(Right(NUMERO, 5))
            End If
        Else
            MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON " & CENTENA(Right(NUMERO, 6))
        End If
    Else
        MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON " & MILLAR1(Right(NUMERO, 8))
    End If
Else
    MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON " & MILLAR2(Right(NUMERO, 9))
End If
Else
    MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON " & MILLAR3(Right(NUMERO, 10))
End If
Else
If Mid(NUMERO, 3, 1) = 0 Then
    If Mid(NUMERO, 4, 1) = 0 Then
        If Mid(NUMERO, 5, 1) = 0 Then
            If Mid(NUMERO, 7, 1) = 0 Then
                If Mid(NUMERO, 8, 1) = 0 Then
                    If Mid(NUMERO, 9, 1) = 0 Then
                        MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES DE "
                    Else

```

```

                                MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES " & UNIDAD(Right(NUMERO,
4))
                                End If
                                Else
                                MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES " & DECENA(Right(NUMERO, 5))
                                End If
                                Else
                                MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES " & CENTENA(Right(NUMERO, 6))
                                End If
                                Else
                                MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES " & MILLAR1(Right(NUMERO, 8))
                                End If
                                Else
                                MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES " & MILLAR2(Right(NUMERO, 9))
                                End If
                                Else
                                MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES " & MILLAR3(Right(NUMERO, 10))
                                End If
                                End If
                                End Function
```

Option Compare Database

Sub Inicio()

TASA_CAMBIO.Visible = False

ID_VENTA.Visible = False

FECHA_CANCEL.Visible = False

ESTADO.Visible = False

VALOR_TOTAL_CORDOBAS.Visible = False

TASA.Locked = True

NOMBRE.Locked = True

VALOR_RECIBO.Locked = True

TOTAL_CORDOBAS.Locked = True

VALOR_LETRA.Locked = True

ID_BANCO.Locked = True

REFERENCIA.Locked = True

TOTAL_CORDOBAS = ""

RECIBO.Visible = False

End Sub

Sub Cancelar()

ESTADO = "CANCELADO"

FECHA_CANCEL = Date

MsgBox "Recibo Cancelado N° " & ID_RECIBO, vbInformation, "SAVS"

DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70

End Sub

Private Sub btnCancelar_Click()

If DOLARES <> "" Then

Else

DOLARES = 0

End If

If CORDOBAS <> "" Then

Else

CORDOBAS = 0

End If

monto = VALOR_RECIBO * TASA

VALOR_TOTAL_CORDOBAS = (DOLARES * TASA) + CORDOBAS

If TIPO_PAGO <> 0 Then

If TIPO_PAGO <> 1 Then

If REFERENCIA <> "" And ID_BANCO <> 0 Then

If monto = VALOR_TOTAL_CORDOBAS Then

If VALOR_LETRA <> "" Then

Cancelar

SALIR

DoCmd.OpenReport "IMPRIMIR_RECIBO_NOEFECTIVO1", acViewPreview

DoCmd.Close acForm, "CANCEL_RECIBO1"

DoCmd.Close acForm, "VENTAS"

Else

MsgBox "Introduzca el Valor en Letras", vbInformation, "SAVS"

End If

Else

MsgBox "El monto recibido no equivale al valor de la cuota", vbInformation, "SAV

S"

End If

Else

MsgBox "Ingrese los datos de la Modalidad de Pago", vbInformation, "SAVS"

End If

Else

If monto = VALOR_TOTAL_CORDOBAS Then

If VALOR_LETRA <> "" Then

Cancelar

```
        SALIR
        DoCmd.OpenReport "IMPRIMIR_RECIBO_EFECTIVO1", acViewPreview
        DoCmd.Close acForm, "CANCEL_RECIBO1"
        DoCmd.Close acForm, "VENTAS"
    Else
        MsgBox "Introduzca el Valor en Letras", vbInformation, "SAVS"
    End If
Else
    MsgBox "El monto recibido no equivale al valor de la cuota", vbInformation, "SAVS"
End If
End If
Else
    MsgBox "Elija la Modalidad de Pago", vbInformation, "SAVS"
End If

End Sub

Private Sub BUSCAR_KeyPress(Keycode As Integer)
If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 45
Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub CORDOBAS_KeyPress(Keycode As Integer)
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc(
"9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub DOLARES_KeyPress(Keycode As Integer)
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc(
"9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub Form_Open(Cancel As Integer)
Inicio
End Sub

Private Sub ID_BANCO_KeyPress(Keycode As Integer)
If Keycode >= 0 And Keycode <= 255 Then
    Keycode = 0
    Beep
End If

End Sub

Private Sub ID_RECIBO_GotFocus()
RECIBO = ID_RECIBO
TASA = Forms!CANCEL_RECIBO1![TASA_CAMBIO].Form![TASA]
TOTAL_CORDOBAS = VALOR_RECIBO * TASA
```

```
DoCmd.RunMacro "COPIA_PEGA"
DoCmd.GoToControl "TIPO_PAGO"
CENTAVOS = "CORDOBAS CON " & Right(NUMERO, 2) & "/100 CENTAVOS"
Select Case Len(NUMERO)
Case 4
    VALOR_LETRA = UNIDAD(NUMERO) & " " & CENTAVOS
Case 5
    VALOR_LETRA = DECENA(NUMERO) & " " & CENTAVOS
Case 6
    VALOR_LETRA = CENTENA(NUMERO) & " " & CENTAVOS
Case 8
    VALOR_LETRA = MILLAR1(NUMERO) & " " & CENTAVOS
Case 9
    VALOR_LETRA = MILLAR2(NUMERO) & " " & CENTAVOS
Case 10
    VALOR_LETRA = MILLAR3(NUMERO) & " " & CENTAVOS
Case 12
    VALOR_LETRA = MILLON1(NUMERO) & " " & CENTAVOS

End Select
End Sub
Function UNIDAD(NUMERO) As String
Select Case Left(Right(NUMERO, 4), 1)
Case 1
    UNIDAD = "UN"
Case 2
    UNIDAD = "DOS"
Case 3
    UNIDAD = "TRES"
Case 4
    UNIDAD = "CUATRO"
Case 5
    UNIDAD = "CINCO"
Case 6
    UNIDAD = "SEIS"
Case 7
    UNIDAD = "SIETE"
Case 8
    UNIDAD = "OCHO"
Case 9
    UNIDAD = "NUEVE"
End Select
End Function
Function DECENA(NUMERO) As String
Select Case Left(Right(NUMERO, 5), 2)
Case 10
    DECENA = "DIEZ "
Case 11
    DECENA = "ONCE "
Case 12
    DECENA = "DOCE "
Case 13
    DECENA = "TRECE "
Case 14
    DECENA = "CATORCE "
Case 15
    DECENA = "QUINCE "
Case 16 To 19
    DECENA = "DIEZ Y " & UNIDAD(Right(NUMERO, 4))
Case 20
    DECENA = "VEIENTE "
```

```

Case 21 To 29
    DECENA = "VEINTE Y " & UNIDAD(Right(NUMERO, 4))
Case 30
    DECENA = "TREINTA " & UNIDAD(Right(NUMERO, 4))
Case 31 To 39
    DECENA = "TREINTA Y " & UNIDAD(Right(NUMERO, 4))
Case 40
    DECENA = "CUARENTA "
Case 41 To 49
    DECENA = "CUARENTA Y " & UNIDAD(Right(NUMERO, 4))
Case 50
    DECENA = "CINCUENTA "
Case 51 To 59
    DECENA = "CINCUENTA Y " & UNIDAD(Right(NUMERO, 4))
Case 60
    DECENA = "SESENTA "
Case 61 To 69
    DECENA = "SESENTA Y " & UNIDAD(Right(NUMERO, 4))
Case 70
    DECENA = "SETENTA"
Case 71 To 79
    DECENA = "SETENTA Y " & UNIDAD(Right(NUMERO, 4))
Case 80
    DECENA = "OCHENTA"
Case 81 To 89
    DECENA = "OCHENTA Y " & UNIDAD(Right(NUMERO, 4))
Case 90
    DECENA = "NOVENTA"
Case 91 To 99
    DECENA = "NOVENTA Y " & UNIDAD(Right(NUMERO, 4))
End Select
End Function

Function CENTENA(NUMERO) As String
Select Case Left(Right(NUMERO, 6), 3)
Case 100
    CENTENA = "CIEN "
Case 101 To 109
    CENTENA = "CIENTO " & UNIDAD(Right(NUMERO, 4))
Case 110 To 199
    CENTENA = "CIENTO " & DECENA(Right(NUMERO, 5))
Case 200, 300, 400, 600, 800
    CENTENA = UNIDAD(Left(NUMERO, 4)) & "CIENTOS "
Case 201 To 209
    CENTENA = "DOSCIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 210 To 299
    CENTENA = "DOSCIENTOS " & DECENA(Right(NUMERO, 5))
Case 301 To 309
    CENTENA = "TRESCIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 310 To 399
    CENTENA = "TRESCIENTOS " & DECENA(Right(NUMERO, 5))
Case 401 To 409
    CENTENA = "CUATROCIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 410 To 499
    CENTENA = "CUATROCIENTOS " & DECENA(Right(NUMERO, 5))
Case 500
    CENTENA = "QUINIENTOS "
Case 501 To 509
    CENTENA = "QUINIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 510 To 599
    CENTENA = "QUINIENTOS " & DECENA(Right(NUMERO, 5))

```

```

Case 601 To 609
    CENTENA = "SEISCIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 610 To 699
    CENTENA = "SEISCIENTOS " & DECENA(Right(NUMERO, 5))
Case 700
    CENTENA = "SETECIENTOS "
Case 701 To 709
    CENTENA = "SETECIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 710 To 799
    CENTENA = "SETECIENTOS " & DECENA(Right(NUMERO, 5))
Case 801 To 809
    CENTENA = "OCHOCIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 810 To 899
    CENTENA = "OCHOCIENTOS " & DECENA(Right(NUMERO, 5))
Case 900
    CENTENA = "NOVECIENTOS "
Case 901 To 909
    CENTENA = "NOVECIENTOS " & UNIDAD(Right(NUMERO, 4))
Case 910 To 999
    CENTENA = "NOVECIENTOS " & DECENA(Right(NUMERO, 5))
End Select
End Function

```

```

Function MILLAR1(NUMERO) As String
If Mid(NUMERO, 3, 1) = 0 Then
    If Mid(NUMERO, 4, 1) = 0 Then
        If Mid(NUMERO, 5, 1) = 0 Then
            MILLAR1 = UNIDAD(Left(NUMERO, 4)) & " MIL"
        Else
            MILLAR1 = UNIDAD(Left(NUMERO, 4)) & " MIL " & UNIDAD(Right(NUMERO, 4))
        End If
    Else
        MILLAR1 = UNIDAD(Left(NUMERO, 4)) & " MIL " & DECENA(Right(NUMERO, 5))
    End If
Else
    MILLAR1 = UNIDAD(Left(NUMERO, 4)) & " MIL " & CENTENA(Right(NUMERO, 6))
End If

```

```

End Function
Function MILLAR2(NUMERO) As String
If Mid(NUMERO, 4, 1) = 0 Then
    If Mid(NUMERO, 5, 1) = 0 Then
        If Mid(NUMERO, 6, 1) = 0 Then
            MILLAR2 = DECENA(Left(NUMERO, 5)) & " MIL"
        Else
            MILLAR2 = DECENA(Left(NUMERO, 5)) & " MIL " & UNIDAD(Right(NUMERO, 4))
        End If
    Else
        MILLAR2 = DECENA(Left(NUMERO, 5)) & " MIL " & DECENA(Right(NUMERO, 5))
    End If
Else
    MILLAR2 = DECENA(Left(NUMERO, 5)) & " MIL " & CENTENA(Right(NUMERO, 6))
End If

```

```

End Function
Function MILLAR3(NUMERO) As String
If Mid(NUMERO, 5, 1) = 0 Then
    If Mid(NUMERO, 6, 1) = 0 Then
        If Mid(NUMERO, 7, 1) = 0 Then
            MILLAR3 = CENTENA(Left(NUMERO, 6)) & " MIL"
        Else
            MILLAR3 = CENTENA(Left(NUMERO, 6)) & " MIL " & UNIDAD(Right(NUMERO, 4))

```

```

        End If
    Else
        MILLAR3 = CENTENA(Left(NUMERO, 6)) & " MIL " & DECENA(Right(NUMERO, 5))
    End If
Else
    MILLAR3 = CENTENA(Left(NUMERO, 6)) & " MIL " & CENTENA(Right(NUMERO, 6))
End If
End Function
Function MILLON1(NUMERO) As String
If Mid(NUMERO, 1, 1) = 1 Then
If Mid(NUMERO, 3, 1) = 0 Then
    If Mid(NUMERO, 4, 1) = 0 Then
        If Mid(NUMERO, 5, 1) = 0 Then
            If Mid(NUMERO, 7, 1) = 0 Then
                If Mid(NUMERO, 8, 1) = 0 Then
                    If Mid(NUMERO, 9, 1) = 0 Then
                        MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON DE "
                    Else
                        MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON " & UNIDAD(Right(NUMERO, 4))
                    )
                End If
            Else
                MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON " & DECENA(Right(NUMERO, 5))
            End If
        Else
            MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON " & CENTENA(Right(NUMERO, 6))
        End If
    Else
        MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON " & MILLAR1(Right(NUMERO, 8))
    End If
Else
    MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON " & MILLAR2(Right(NUMERO, 9))
End If
Else
    MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLON " & MILLAR3(Right(NUMERO, 10))
End If
Else
If Mid(NUMERO, 3, 1) = 0 Then
    If Mid(NUMERO, 4, 1) = 0 Then
        If Mid(NUMERO, 5, 1) = 0 Then
            If Mid(NUMERO, 7, 1) = 0 Then
                If Mid(NUMERO, 8, 1) = 0 Then
                    If Mid(NUMERO, 9, 1) = 0 Then
                        MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES DE "
                    Else
                        MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES " & UNIDAD(Right(NUMERO,
4))
                    )
                End If
            Else
                MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES " & DECENA(Right(NUMERO, 5))
            End If
        Else
            MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES " & CENTENA(Right(NUMERO, 6))
        End If
    Else
        MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES " & MILLAR1(Right(NUMERO, 8))
    End If
Else
    MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES " & MILLAR2(Right(NUMERO, 9))
End If
Else

```

Form_CANCEL_RECIBO1 - 7

```
MILLON1 = UNIDAD(Left(NUMERO, 4)) & " MILLONES " & MILLAR3(Right(NUMERO, 10))  
End If
```

```
End If  
End Function
```

```
End Function
```

```
Private Sub TASA_KeyPress(Keycode As Integer)
```

```
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 45 Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
```

```
Else  
    Keycode = 0  
    Beep
```

```
End If
```

```
End Sub
```

```
Private Sub TIPO_PAGO_AfterUpdate()
```

```
If TIPO_PAGO <> 1 Then  
    REFERENCIA.Locked = False  
    ID_BANCO.Locked = False
```

```
Else  
    REFERENCIA = ""  
    ID_BANCO = ""  
    REFERENCIA.Locked = True  
    ID_BANCO.Locked = True
```

```
End If
```

```
End Sub
```

```
Private Sub TIPO_PAGO_KeyPress(Keycode As Integer)
```

```
If Keycode >= 0 And Keycode <= 255 Then  
    Keycode = 0  
    Beep
```

```
End If
```

```
End Sub
```

```
Sub SALIR()
```

```
Form_VENTAS.FECHA_ADQUI.Locked = False  
Form_VENTAS.ID_CLIENTE.Locked = False  
Form_VENTAS.ID_POL.Locked = False  
Form_VENTAS.ID_PROMOTOR.Locked = False  
Form_VENTAS.FORMA_PAGO.Locked = False  
Form_VENTAS.btnGuardar.Enabled = True  
Form_VENTAS.btnSalir.Enabled = True
```

```
End Sub
```

Option Compare Database

```
Private Sub Form_Open(Cancel As Integer)
```

```
DoCmd.GoToRecord , , acNewRec
```

```
End Sub
```

```
Private Sub NOMBRE_KeyPress(Keycode As Integer)
```

```
If Keycode = Asc("á") Or Keycode = Asc("é") Or Keycode = Asc("í") Or Keycode = Asc("ó") Or Keycode = Asc("ú") Or (Keycode >= 97 And Keycode <= 122) Or (Keycode >= 65 And Keycode <= 90) Or Keycode = 8 Or Keycode = 32 Or Keycode = 209 Or Keycode = 13 Or Keycode = 241 Then
```

```
Else
```

```
Keycode = 0
```

```
Beep
```

```
End If
```

```
End Sub
```

```
Private Sub TIPO_ID_KeyPress(Keycode As Integer)
```

```
If Keycode >= 0 And Keycode <= 255 Then
```

```
Keycode = 0
```

```
Beep
```

```
End If
```

```
End Sub
```

```
Private Sub btnGuardar_Click()
```

```
BAN = 0
```

```
DoCmd.GoToRecord , , acLast
```

```
If NOMBRE <> "" And DIRECCION <> "" And FECHA_VISITA <> "" And TIPO_ID <> 0 And NUM_ID <> "" Then
```

```
var = NOMBRE
```

```
HASTA = Form_CLIENES.CurrentRecord
```

```
If HASTA = 1 Then
```

```
DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
```

```
DoCmd.GoToRecord , , acNewRec
```

```
Else
```

```
For I = HASTA - 1 To 1 Step -1
```

```
DoCmd.GoToRecord , , acPrevious
```

```
If var = NOMBRE Then
```

```
MsgBox "Cliente existente", vbInformation, "SAVS"
```

```
DoCmd.GoToRecord , , acLast
```

```
I = 0
```

```
BAN = 1
```

```
End If
```

```
Next I
```

```
If BAN = 0 Then
```

```
DoCmd.GoToRecord , , acGoTo, HASTA
```

```
DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
```

```
cotizacion = MsgBox("¿Desea hacer una Cotización?", vbYesNo, "SAVS")
```

```
If cotizacion = vbYes Then
```

```
DoCmd.OpenForm "COTIZACIONES", , , stLinkCriteria
```

```
Else
```

```
DoCmd.GoToRecord , , acNewRec
```

```
End If
```

```
End If
```

```
End If
```

```
Else
```

```
MsgBox "Ingrese los datos del Cliente", vbInformation, "SAVS"
```

```
End If
```

```
End Sub
```

```
Private Sub btnSalir_Click()
```

```
If ID_CLIENTE <> "" Then
```

Form_CLIENES - 2

```
    If NOMBRE <> "" And DIRECCION <> "" And FECHA_VISITA <> "" And TIPO_ID <> 0 And NUM_ID <> ""
Then
    MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"
Else
    respuesta = MsgBox("Datos no guardados, ¿Desea Salir?", vbYesNo, "SAVS")
    If respuesta = vbYes Then
        DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
        DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
        DoCmd.Close
    End If
End If
Else
    DoCmd.Close
End If
End Sub
```

Form_COBERTURAS - 1

Option Compare Database

```
Private Sub Form_Open(Cancel As Integer)
ID_POL.Locked = True
```

```
DoCmd.GoToRecord , , acNewRec
```

```
End Sub
```

```
Private Sub btnGuardar_Click()
```

```
BAN = 0
```

```
DoCmd.GoToRecord , , acLast
```

```
If COBERTURA <> "" And VALOR_COBER <> 0 Then
```

```
    var = COBERTURA
```

```
    HASTA = Form_COBERTURAS.CurrentRecord
```

```
    If HASTA = 1 Then
```

```
        Guardar
```

```
    Else
```

```
        For I = HASTA - 1 To 1 Step -1
```

```
            DoCmd.GoToRecord , , acPrevious
```

```
            If var = COBERTURA Then
```

```
                MsgBox "Cobertura existente", vbInformation, "SAVS"
```

```
                DoCmd.GoToRecord , , acLast
```

```
                I = 0
```

```
                BAN = 1
```

```
            End If
```

```
        Next I
```

```
        If BAN = 0 Then
```

```
            Guardar
```

```
        End If
```

```
    End If
```

```
Else
```

```
    MsgBox "Ingrese los datos de la Cobertura", vbInformation, "SAVS"
```

```
End If
```

```
End Sub
```

```
Sub Guardar()
```

```
DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
```

```
DoCmd.GoToRecord , , acNewRec
```

```
Form_COBERTURAS.ID_POL = Form_POLIZA_COMPRA.ID_POL
```

```
End Sub
```

```
Private Sub btnSalir_Click()
```

```
If Form_COBERTURAS.CurrentRecord > 1 Then
```

```
If COBERTURA <> "" And VALOR_COBER <> 0 Then
```

```
    MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"
```

```
Else
```

```
    If COBERTURA <> "" Or VALOR_COBER <> 0 Then
```

```
        respuesta = MsgBox("Datos no guardados, ¿Desea Salir?", vbYesNo, "SAVS")
```

```
        If respuesta = vbYes Then
```

```
            DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
```

```
            DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
```

```
            Cerrar_Abrir
```

```
        End If
```

```
    Else
```

```
        DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
```

```
        DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
```

```
        Cerrar_Abrir
```

```
    End If
```

```
End If
```

```
Else
```

```
MsgBox "Toda Póliza debe tener al menos una Cobertura", vbInformation, "SAVS"
```

```
End If
```

```
End Sub
```

Form_COBERTURAS - 2

Sub Cerrar_Abrir()

Dim stLinkCriteria As String

stLinkCriteria = "[ID_POL]=" & "'" & Me![ID_POL] & "'"

DoCmd.Close

DoCmd.OpenForm "CUOTAS", , , stLinkCriteria

End Sub

Private Sub VALOR_COBER_KeyPress(Keycode As Integer)

If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 45 Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then

Else

 Keycode = 0

 Beep

End If

End Sub

Option Compare Database

Private Sub btnSalir_Click()

 DoCmd.Close

End Sub

Private Sub BUSCAR_AfterUpdate()

 ' Buscar el registro que coincida con el control.

 Dim rs As Object

 Set rs = Me.Recordset.Clone

 rs.FindFirst "[ID_COTIZACION] = " & Str(Me![BUSCAR])

 Me.Bookmark = rs.Bookmark

 If ID_COTIZACION = BUSCAR Then

 Else

 DoCmd.GoToRecord , , acNewRec

 MsgBox "Esta Cotización no existe", vbInformation, "SAVS"

 BUSCAR = ""

 DoCmd.GoToControl "BUSCAR"

 End If

End Sub

Private Sub BUSCAR_KeyPress(Keycode As Integer)

 If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13

 Or Keycode = 28 Or Keycode = 9 Then

 Else

 Keycode = 0

 Beep

 End If

End Sub

Private Sub Form_Open(Cancel As Integer)

 ID_COTIZACION.Visible = False

 FECHA.Visible = False

 CLIENTE.Locked = True

 ID_POL.Locked = True

 CONTACTO.Locked = True

 ID_PROMOTOR.Locked = True

 DoCmd.GoToRecord , , acNewRec

End Sub

Private Sub btnAceptar_Click()

 If ID_CLIENTE <> 0 And ID_POL <> 0 And ID_PROMOTOR <> 0 Then

 DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70

 DoCmd.OpenReport "COTIZACION_IMPRIMIR", acViewPreview

 DoCmd.Close acForm, "COTIZACION_IMPRIMIR"

 Else

 MsgBox "Complete los datos de la Cotización", vbInformation, "SAVS"

 End If

End Sub

Private Sub Cuadro_combinado30_AfterUpdate()

 ' Buscar el registro que coincida con el control.

 Dim rs As Object

Form_COTIZACION_IMPRIMIR - 2

```
Set rs = Me.Recordset.Clone
rs.FindFirst "[ID_COTIZACION] = " & Str(Me![Cuadro combinado30])
Me.Bookmark = rs.Bookmark
End Sub
```

Option Compare Database

```
Private Sub btnSalir_Click()  
If IsNull(ID_COTIZACION) = True Then  
    DoCmd.Close  
Else  
    respuesta = MsgBox("La Cotización no se generará, ¿Desea Salir?", vbYesNo, "SAVS")  
    If respuesta = vbYes Then  
        DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70  
        DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70  
        DoCmd.Close  
    End If  
End If  
  
End Sub
```

```
Private Sub Form_Open(Cancel As Integer)  
DoCmd.GoToRecord , , acNewRec  
FECHA.Locked = True  
End Sub  
  
Private Sub btnAceptar_Click()  
If ID_CLIENTE <> 0 And ID_POL <> 0 And ID_PROMOTOR <> 0 Then  
    DoCmd.DoMenuItem acFormBar, acRecordsMenu, 5, , acMenuVer70  
    DoCmd.OpenReport "COTIZACION_NUEVA", acViewPreview  
    DoCmd.Close acForm, "COTIZACION_NUEVA"  
Else  
    MsgBox "Complete los datos de la Cotización", vbInformation, "SAVS"  
End If  
  
End Sub
```

Option Compare Database

```
Private Sub btnSalir_Click()
If CONTACTO <> "" And POLIZA <> 0 And PROMOTOR <> 0 Then
    MsgBox "Debe guardar antes de Salir", vbInformation, "SAVS"
Else
    Form_CLIENES.ID_CLIENES.Locked = False
    Form_CLIENES.FECHA_VISITA.Locked = False
    Form_CLIENES.NOMBRE.Locked = False
    Form_CLIENES.DIRECCION.Locked = False
    Form_CLIENES.TELEFONO.Locked = False
    Form_CLIENES.CELULAR.Locked = False
    Form_CLIENES.TIPO_ID.Locked = False
    Form_CLIENES.NUM_ID.Locked = False
    DoCmd.Close
    DoCmd.GoToRecord , , acNewRec
End If
End Sub

Private Sub Form_Open(Cancel As Integer)
DoCmd.GoToRecord , , acNewRec
Form_CLIENES.ID_CLIENES.Locked = True
Form_CLIENES.FECHA_VISITA.Locked = True
Form_CLIENES.NOMBRE.Locked = True
Form_CLIENES.DIRECCION.Locked = True
Form_CLIENES.TELEFONO.Locked = True
Form_CLIENES.CELULAR.Locked = True
Form_CLIENES.TIPO_ID.Locked = True
Form_CLIENES.NUM_ID.Locked = True
ID_CLIENES.Visible = False
PARAMETRO.Visible = False
End Sub

Private Sub POLIZA_AfterUpdate()
Form_COTIZACIONES.ID_CLIENES = Form_CLIENES.ID_CLIENES
End Sub

Private Sub POLIZA_KeyPress(Keycode As Integer)
If Keycode >= 0 And Keycode <= 255 Then
    Keycode = 0
    Beep
End If

End Sub

Private Sub btnGuardar_Click()
If CONTACTO <> "" And POLIZA <> 0 And PROMOTOR <> 0 Then
    PARAMETRO = ID_COTIZACION
    DoCmd.GoToRecord , , acNewRec
    DoCmd.OpenReport "COTIZACION", acViewPreview
Else
    MsgBox "Complete los datos de la Cotización", vbInformation, "SAVS"
End If
End Sub
```

Option Compare Database

```

Private Sub Form_Open(Cancel As Integer)
DoCmd.GoToRecord , , acNewRec
ID_POL.Locked = True
VALOR_CUOTA.Locked = True
CANT_PAGOS.Visible = False
VALOR_VENTA.Visible = False
End Sub
Private Sub btnGuardar_Click()
BAN = 0
DoCmd.GoToRecord , , acLast
If FORMA_PAGO <> 0 Then
    var = FORMA_PAGO
    HASTA = Form_CUOTAS.CurrentRecord
    If HASTA = 1 Then
        DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
        DoCmd.GoToRecord , , acNewRec
    Else
        For I = HASTA - 1 To 1 Step -1
            DoCmd.GoToRecord , , acPrevious
            If var = FORMA_PAGO Then
                MsgBox "Forma de Pago existente", vbInformation, "SAVS"
                DoCmd.GoToRecord , , acLast
                I = 0
                BAN = 1
            End If
        Next I
        If BAN = 0 Then
            DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
            DoCmd.GoToRecord , , acNewRec
        End If
    End If
Else
    MsgBox "Ingrese los datos de la Cuota", vbInformation, "SAVS"
End If

End Sub
Private Sub btnSalir_Click()
If Form_CUOTAS.CurrentRecord > 1 Then
If FORMA_PAGO <> 0 Then
    MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"
Else
    If FORMA_PAGO <> 0 Then
        respuesta = MsgBox("Datos no guardados, ¿Desea Salir?", vbYesNo, "SAVS")
        If respuesta = vbYes Then
            DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
            DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
            SALIR
        End If
    Else
        SALIR
    End If
End If
Else
    MsgBox "Toda Póliza debe tener al menos una Forma de Pago", vbInformation, "SAVS"
End If
End Sub
Sub SALIR()
Form_POLIZA_COMPRA.ID_PROVEEDOR.Locked = False

```

Form_CUOTAS - 2

```
Form_POLIZA_COMPRA.ID_POL.Locked = False
Form_POLIZA_COMPRA.DESCRIPCION.Locked = False
Form_POLIZA_COMPRA.VALOR_COSTO.Locked = False
Form_POLIZA_COMPRA.CANT_BENEFIC.Locked = False
Form_POLIZA_COMPRA.VALOR_VENTA.Locked = False
Form_POLIZA_COMPRA.PORC_COMISION.Locked = False
Form_POLIZA_COMPRA.btnGuardar.Enabled = True
Form_POLIZA_COMPRA.btnSalir.Enabled = True
Form_POLIZA_COMPRA.RAM = ""
DoCmd.Close
DoCmd.GoToRecord , , acNewRec
End Sub
```

```
Private Sub FORMA_PAGO_AfterUpdate()
ID_POL = Form_POLIZA_COMPRA.ID_POL
VALOR_CUOTA = VALOR_VENTA / CANT_PAGOS
End Sub
```

```
Private Sub FORMA_PAGO_KeyPress(Keycode As Integer)
If Keycode >= 0 And Keycode <= 255 Then
    Keycode = 0
    Beep
End If
End Sub
```

Option Compare Database

```
Private Sub btnSalir_Click()  
DoCmd.Close  
End Sub  
Private Sub btnEjecutar_Click()  
If OPCION = 1 Then  
    If BUSCAR_CLIENTE <> 0 Then  
        DoCmd.OpenReport "ESTADO_CUENTA_CLIENTE", acViewPreview  
    Else  
        MsgBox "Seleccione el Cliente", vbInformation, "SAVS"  
    End If  
Else  
    If OPCION = 2 Then  
        If ID_VENTA <> "" Then  
            DoCmd.OpenReport "ESTADO_CUENTA_VENTA", acViewPreview  
        Else  
            MsgBox "Ingrese el Número de la Venta", vbInformation, "SAVS"  
        End If  
    Else  
        MsgBox "Seleccione el tipo de Estado de Cuenta", vbInformation, "SAVS"  
    End If  
End If  
End Sub  
Private Sub Form_Open(Cancel As Integer)  
BUSCAR_CLIENTE.Enabled = False  
ID_VENTA.Enabled = False  
End Sub  
Private Sub ID_VENTA_KeyPress(Keycode As Integer)  
If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13  
Or Keycode = 28 Or Keycode = 9 Then  
Else  
    Keycode = 0  
    Beep  
End If  
End Sub  
Private Sub OPCION_AfterUpdate()  
If OPCION = 1 Then  
    ID_VENTA = ""  
    BUSCAR_CLIENTE.Enabled = True  
    ID_VENTA.Enabled = False  
Else  
    If OPCION = 2 Then  
        BUSCAR_CLIENTE = ""  
        BUSCAR_CLIENTE.Enabled = False  
        ID_VENTA.Enabled = True  
    End If  
End If  
End Sub
```

Option Compare Database

```

Private Sub btnGuardar_Click()
BAN = 0
DoCmd.GoToRecord , , acLast
If DESCRPCION <> "" And CANT_PAGOS <> 0 Then
    var = DESCRIPCION
    HASTA = Form_FORMA_PAGO.CurrentRecord
    If HASTA = 1 Then
        DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
        DoCmd.GoToRecord , , acNewRec
    Else
        For I = HASTA - 1 To 1 Step -1
            DoCmd.GoToRecord , , acPrevious
            If var = DESCRIPCION Then
                MsgBox "Forma de pago existente", vbInformation, "SAVS"
                DoCmd.GoToRecord , , acLast
                I = 0
                BAN = 1
            End If
        Next I
        If BAN = 0 Then
            DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
            DoCmd.GoToRecord , , acNewRec
        End If
    End If
Else
    MsgBox "Ingrese los datos de la Forma de Pago", vbInformation, "SAVS"
End If

End Sub

Private Sub btnSalir_Click()

If FORMA_PAGO <> "" Then
    If DESCRIPCION <> "" And CANT_PAGOS <> 0 Then
        MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"
    Else
        respuesta = MsgBox("Datos no guardados, ¿Desea Salir?", vbYesNo, "SAVS")
        If respuesta = vbYes Then
            DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
            DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
            DoCmd.Close
        End If
    End If
Else
    DoCmd.Close
End If

End Sub

Private Sub CANT_PAGOS_KeyPress(Keycode As Integer)
If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13
Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub DESCRIPCION_KeyPress(Keycode As Integer)

```

Form_FORMA_PAGO - 2

```
If Keycode = Asc("á") Or Keycode = Asc("é") Or Keycode = Asc("í") Or Keycode = Asc("ó") Or Keycode = Asc("ú") Or (Keycode >= 97 And Keycode <= 122) Or (Keycode >= 65 And Keycode <= 90) Or Keycode = 8 Or Keycode = 32 Or Keycode = 209 Or Keycode = 13 Or Keycode = 241 Then
```

```
Else
```

```
    Keycode = 0
```

```
    Beep
```

```
End If
```

```
End Sub
```

```
Private Sub Form_Open(Cancel As Integer)
```

```
DoCmd.GoToRecord , , acNewRec
```

```
End Sub
```

```
Private Sub Form_Open(Cancel As Integer)
NUMERO.Enabled = False
FECHA1.Enabled = False
FECHA2.Enabled = False
End Sub

Private Sub NUMERO_KeyPress(Keycode As Integer)
If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13
Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If
End Sub

Private Sub OPCION_AfterUpdate()
If OPCION = 1 Then
    FECHA1 = ""
    FECHA2 = ""
    NUMERO.Enabled = True
    FECHA1.Enabled = False
    FECHA2.Enabled = False
Else
    NUMERO = ""
    NUMERO.Enabled = False
    FECHA1.Enabled = True
    FECHA2.Enabled = True
End If
End Sub

Private Sub btnEjecutar_Click()
If OPCION = 1 Then
    DoCmd.OpenReport "REIMPRIMIR_COBRO_NUMERO", acViewPreview
Else
    If OPCION = 2 Then
        DoCmd.OpenReport "IMPRIMIR_COBROS", acViewPreview
    Else
        MsgBox "Elija las opciones necesarias", vbInformation, "SAVS"
    End If
End If
End Sub

Private Sub btnSalir_Click()
DoCmd.Close
End Sub
```

Form_INFO_EMPRESA - 1

Option Compare Database

```
Private Sub btnSalir_Click()  
    DoCmd.Close
```

```
End Sub
```

Option Compare Database

```
Private Sub Form_Open(Cancel As Integer)
```

```
FECHA.Locked = True
```

```
DoCmd.GoToControl "TASA"
```

```
DoCmd.GoToRecord , , acLast
```

```
If FECHA <> Date Then
```

```
    DoCmd.GoToRecord , , acNewRec
```

```
Else
```

```
    DoCmd.Close
```

```
End If
```

```
End Sub
```

```
Private Sub btnGuardar_Click()
```

```
If TASA <> "" Then
```

```
    If TASA > 0 Then
```

```
        DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
```

```
        DoCmd.Close
```

```
    Else
```

```
        MsgBox "Tasa de Cambio Inválida", vbInformation, "SAVS"
```

```
TASA = 0
```

```
    End If
```

```
Else
```

```
    MsgBox "Ingrese la Tasa de Cambio de Hoy", vbInformation, "SAVS"
```

```
End If
```

```
End Sub
```

```
Private Sub TASA_KeyPress(Keycode As Integer)
```

```
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
```

```
Else
```

```
    Keycode = 0
```

```
    Beep
```

```
End If
```

```
End Sub
```

Form_INICIO - 1

Option Compare Database

```
Private Sub Form_Open(Cancel As Integer)
EliminaBotonCerrar Me
PARAMETRO.Visible = False
FECHA_MAX.Visible = False
PARAMETRO = ""
PARAMETRO = FECHA_MAX + 1
While PARAMETRO < Now()
    DoCmd.OpenQuery "RECIBOS_LOTE", acViewNormal, acEdit
    PARAMETRO = PARAMETRO + 1
Wend
DoCmd.OpenQuery "DESACTIVADO", acViewNormal, acEdit
DoCmd.Close
DoCmd.OpenForm "INGRESAR_TASA", , , stLinkCriteria
End Sub
```

Form_LISTADO_CLIENTES - 1

Option Compare Database

Private Sub btnEjecutar_Click()

If FECHA1 <> "" And FECHA2 <> "" And OPCION <> 0 Then

 If OPCION = 1 Then

 DoCmd.OpenReport "CLIENTES_ACTIVOS", acViewPreview

 Else

 DoCmd.OpenReport "CLIENTES_INACTIVOS", acViewPreview

 End If

Else

 MsgBox "Ingrese los parámetros necesarios", vbInformation, "SAVS"

End If

End Sub

Private Sub btnSalir_Click()

 DoCmd.Close

End Sub

Form_MODIFICAR_BANCO - 1

Option Compare Database
Public BANCO1 As String

```
Private Sub btnGuardar_Click()  
BAN = 0  
var = BANCO  
pos = Form_MODIFICAR_BANCO.CurrentRecord  
DoCmd.GoToRecord , , acLast  
If BANCO <> "" Then  
    HASTA = Form_MODIFICAR_BANCO.CurrentRecord  
    For I = HASTA To 1 Step -1  
        If var = BANCO And pos <> Form_MODIFICAR_BANCO.CurrentRecord Then  
            MsgBox "Banco existente", vbInformation, "SAVS"  
            DoCmd.GoToRecord , , acGoTo, pos  
            BANCO = BANCO1  
            I = 0  
            BAN = 1  
        End If  
        If Form_MODIFICAR_BANCO.CurrentRecord <> 1 Then  
            DoCmd.GoToRecord , , acPrevious  
        End If  
    Next I  
    DoCmd.GoToRecord , , acGoTo, pos  
    If BAN = 0 Then  
        MsgBox "Banco Modificado", vbInformation, "SAVS"  
    End If  
End If
```

```
Else  
    MsgBox "Ingrese el nombre del Banco", vbInformation, "SAVS"  
End If
```

```
BANCO.Locked = True  
BUSCAR_BANCO.Locked = False  
End Sub
```

```
Private Sub btnSalir_Click()
```

```
If BANCO.Locked = True Then  
    DoCmd.Close
```

```
Else  
    MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"  
End If
```

```
End Sub
```

```
Private Sub BUSCAR_BANCO_KeyPress(Keycode As Integer)
```

```
If Keycode >= 0 And Keycode <= 255 Then  
    Keycode = 0  
    Beep
```

```
End If
```

```
End Sub
```

```
Private Sub Form_Open(Cancel As Integer)
```

```
DoCmd.GoToRecord , , acNewRec  
BANCO.Locked = True
```

```
End Sub
```

```
Private Sub BUSCAR_BANCO_AfterUpdate()
```

```
    ' Buscar el registro que coincida con el control.
```

```
    Dim rs As Object
```

```
    Set rs = Me.Recordset.Clone  
    rs.FindFirst "[ID_BANCO] = " & Str(Me![BUSCAR_BANCO])
```

```
Me.Bookmark = rs.Bookmark
BANCO1 = BANCO
BANCO.Locked = False
DoCmd.GoToControl "BANCO"
BUSCAR_BANCO.Locked = True
End Sub
```

Form_MODIFICAR_BENEFICIARIO - 1

Option Compare Database

Public POSINI As Integer

Public BENEFICIARIO1 As String

Private Sub BUSCAR_AfterUpdate()

Dim rs As Object

Set rs = Me.Recordset.Clone

rs.FindFirst "[ID_VENTA] = " & Str(Me![BUSCAR])

Me.Bookmark = rs.Bookmark

DoCmd.Requery "BUSCAR_BENEFICIARIO"

BENEFICIARIO.Locked = False

BUSCAR_BENEFICIARIO = ""

If ID_VENTA = BUSCAR Then

POSINI = Form_MODIFICAR_BENEFICIARIO.CurrentRecord

BENEFICIARIO1 = BENEFICIARIO

Else

 Inicio

 MsgBox "Esta Venta no existe", vbInformation, "SAVS"

End If

Inicio

End Sub

Private Sub BUSCAR_BENEFICIARIO_AfterUpdate()

Dim rs As Object

 Set rs = Me.Recordset.Clone

 rs.FindFirst "[BENEFICIARIO] = '" & Me![BUSCAR_BENEFICIARIO] & "'"

 Me.Bookmark = rs.Bookmark

BENEFICIARIO1 = BENEFICIARIO

End Sub

Sub Inicio()

DoCmd.GoToRecord , , acNewRec

ID_VENTA.Visible = False

BENEFICIARIO.Locked = True

BUSCAR = ""

BUSCAR_BENEFICIARIO = ""

End Sub

Private Sub BUSCAR_BENEFICIARIO_KeyPress(Keycode As Integer)

If Keycode >= 0 And Keycode <= 255 Then

 Keycode = 0

 Beep

End If

End Sub

Private Sub BUSCAR_KeyPress(Keycode As Integer)

If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 45
Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then

Else

 Keycode = 0

 Beep

End If

End Sub

Private Sub Form_Open(Cancel As Integer)

Inicio

End Sub

Private Sub btnGuardar_Click()

```

If BUSCAR <> "" Then
BAN = 0
var = BENEFICIARIO
pos = Form_MODIFICAR_BENEFICIARIO.CurrentRecord
DoCmd.GoToRecord , , acGoTo, POSINI
If BENEFICIARIO <> "" Then
    For I = 1 To BUSCAR_BENEFICIARIO.ListCount
        If var = BENEFICIARIO And pos <> Form_MODIFICAR_BENEFICIARIO.CurrentRecord Then
            MsgBox "Beneficiario existente", vbInformation, "SAVS"
            DoCmd.GoToRecord , , acGoTo, pos
            BENEFICIARIO = BENEFICIARIO1
            I = BUSCAR_BENEFICIARIO.ListCount + 1
            BAN = 1
        End If
        If Form_MODIFICAR_BENEFICIARIO.CurrentRecord <> POSINI + BUSCAR_BENEFICIARIO.ListCount -
1 Then
            DoCmd.GoToRecord , , acNext
        End If
    Next I
    Inicio
    If BAN = 0 Then
        MsgBox "Beneficiario Modificado", vbInformation, "SAVS"
    End If
Else
    MsgBox "Ingrese el nombre del Beneficiario", vbInformation, "SAVS"
End If
Else
    MsgBox "Ingrese el número de la Venta", vbInformation, "SAVS"
End If
End Sub
Private Sub btnEliminar_Click()
If BUSCAR <> "" Then
If BUSCAR_BENEFICIARIO.ListCount > 1 Then
    respuesta = MsgBox("¿Está seguro de eliminar este Beneficiario?", vbYesNo, "SAVS")
    If respuesta = vbYes Then
        DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
        DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
        Inicio
    End If
Else
    MsgBox "Unico Beneficiario, debe tener al menos uno", vbInformation, "SAVS"
End If
Else
    MsgBox "Ingrese el número de la Venta", vbInformation, "SAVS"
End If
End Sub
Private Sub btnSalir_Click()
If BENEFICIARIO.Locked = True Then
    DoCmd.Close
Else
    MsgBox "Debe guardar antes de Salir", vbInformation, "SAVS"
End If
End Sub

```

Form_MODIFICAR_CLIENTE - 1

Option Compare Database

Public NOMBRE1

Private Sub BUSCAR_CLIENTE_AfterUpdate()

Dim rs As Object

If BUSCAR_CLIENTE <> 0 Then

Set rs = Me.Recordset.Clone

rs.FindFirst "[ID_CLIENTE] = " & Str(Me![BUSCAR_CLIENTE])

Me.Bookmark = rs.Bookmark

FECHA_VISITA.Locked = False

NOMBRE.Locked = False

DIRECCION.Locked = False

TELEFONO.Locked = False

CELULAR.Locked = False

TIPO_ID.Locked = False

NUM_ID.Locked = False

BUSCAR_CLIENTE.Locked = True

NOMBRE1 = NOMBRE

Else

MsgBox "Debe buscar al Cliente", vbInformation, "SAVS"

DoCmd.GoToControl "ID_CLIENTE"

DoCmd.GoToControl "BUSCAR_CLIENTE"

End If

End Sub

Private Sub Form_Open(Cancel As Integer)

FECHA_VISITA.Locked = True

NOMBRE.Locked = True

DIRECCION.Locked = True

TELEFONO.Locked = True

CELULAR.Locked = True

TIPO_ID.Locked = True

NUM_ID.Locked = True

BUSCAR_CLIENTE.Locked = False

DoCmd.GoToRecord , , acNewRec

End Sub

Private Sub NOMBRE_KeyPress(Keycode As Integer)

If Keycode = Asc("á") Or Keycode = Asc("é") Or Keycode = Asc("í") Or Keycode = Asc("ó") Or Keycode = Asc("ú") Or (Keycode >= 97 And Keycode <= 122) Or (Keycode >= 65 And Keycode <= 90) Or Keycode = 8 Or Keycode = 32 Or Keycode = 209 Or Keycode = 13 Or Keycode = 241 Then

Else

Keycode = 0

Beep

End If

End Sub

Private Sub TIPO_ID_KeyPress(Keycode As Integer)

If Keycode >= 0 And Keycode <= 255 Then

Keycode = 0

Beep

End If

End Sub

Private Sub btnGuardar_Click()

If BUSCAR_CLIENTE <> "" Then

BAN = 0

var = NOMBRE

pos = Form_MODIFICAR_CLIENTE.CurrentRecord

DoCmd.GoToRecord , , acLast

If NOMBRE <> "" And DIRECCION <> "" And FECHA_VISITA <> "" And TIPO_ID <> 0 And NUM_ID <> "" The

n

```

HASTA = Form_MODIFICAR_CLIENTE.CurrentRecord
For I = HASTA To 1 Step -1
    If var = NOMBRE And pos <> Form_MODIFICAR_CLIENTE.CurrentRecord Then
        MsgBox "Cliente existente", vbInformation, "SAVS"
        DoCmd.GoToRecord , , acGoTo, pos
        NOMBRE = NOMBRE1
        I = 0
        BAN = 1
    End If
    If Form_MODIFICAR_CLIENTE.CurrentRecord <> 1 Then
        DoCmd.GoToRecord , , acPrevious
    End If
Next I
DoCmd.GoToRecord , , acGoTo, pos
If BAN = 0 Then
    MsgBox "Cliente Guardado", vbInformation, "SAVS"
    DoCmd.GoToRecord , , acNewRec
    BUSCAR_CLIENTE = ""

End If
Else
    MsgBox "Ingrese los datos del Cliente", vbInformation, "SAVS"
End If
FECHA_VISITA.Locked = True
NOMBRE.Locked = True
DIRECCION.Locked = True
TELEFONO.Locked = True
CELULAR.Locked = True
TIPO_ID.Locked = True
NUM_ID.Locked = True
BUSCAR_CLIENTE.Locked = False
Else
    MsgBox "Ingrese el Código del Cliente", vbInformation, "SAVS"
End If
End Sub
Private Sub btnSalir_Click()
    If NOMBRE.Locked = True Then
        DoCmd.Close
    Else
        MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"
    End If
End Sub

Private Sub btnDeshacer_Click()
    On Error GoTo Err_btnDeshacer_Click
    DoCmd.DoMenuItem acFormBar, acEditMenu, acUndo, , acMenuVer70
    DoCmd.GoToRecord , , acNewRec
    BUSCAR_CLIENTE = ""
    FECHA_VISITA.Locked = True
    NOMBRE.Locked = True
    DIRECCION.Locked = True
    TELEFONO.Locked = True
    CELULAR.Locked = True
    TIPO_ID.Locked = True
    NUM_ID.Locked = True
    BUSCAR_CLIENTE.Locked = False
Exit_btnDeshacer_Click:

Exit Sub

```

Err_btnDeshacer_Click:

```
MsgBox "No hay cambios que deshacer", vbInformation, "SAVS"  
DoCmd.GoToRecord , , acNewRec  
BUSCAR_CLIENTE = ""  
FECHA_VISITA.Locked = True  
NOMBRE.Locked = True  
DIRECCION.Locked = True  
TELEFONO.Locked = True  
CELULAR.Locked = True  
TIPO_ID.Locked = True  
NUM_ID.Locked = True  
BUSCAR_CLIENTE.Locked = False  
Resume Exit_btnDeshacer_Click
```

End Sub

Form_MODIFICAR_COBERTURA - 1

```
Option Compare Database
Public COBERTURA1 As String
Public VALOR_COBER1 As Double
Public POSINI As Integer
Sub Inicio()
DoCmd.GoToRecord , , acNewRec
ID_POL.Visible = False
COBERTURA.Locked = True
VALOR_COBER.Locked = True
BUSCAR = ""
BUSCAR_COBERTURA = ""
End Sub
Private Sub btnDeshacer_Click()
On Error GoTo Err_btnDeshacer_Click
DoCmd.DoMenuItem acFormBar, acEditMenu, acUndo, , acMenuVer70
Inicio
Exit_btnDeshacer_Click:

Exit Sub
Err_btnDeshacer_Click:
MsgBox "No hay cambios que deshacer", vbInformation, "SAVS"
Inicio
Resume Exit_btnDeshacer_Click

End Sub

Private Sub BUSCAR_AfterUpdate()
' Buscar el registro que coincida con el control.
Dim rs As Object

Set rs = Me.Recordset.Clone
rs.FindFirst "[ID_POL] = '" & Me![BUSCAR] & "'"
Me.Bookmark = rs.Bookmark
DoCmd.Requery "BUSCAR_COBERTURA"
COBERTURA.Locked = False
VALOR_COBER.Locked = False
BUSCAR_COBERTURA = ""
If ID_POL = BUSCAR Then
POSINI = Form_MODIFICAR_COBERTURA.CurrentRecord
COBERTURA1 = COBERTURA
VALOR_COBER1 = VALOR_COBER
Else
MsgBox "Esta Póliza no existe", vbInformation, "SAVS"
Inicio
End If
End Sub

Private Sub BUSCAR_COBERTURA_AfterUpdate()
Dim rs As Object

Set rs = Me.Recordset.Clone
rs.FindFirst "[COBERTURA] = '" & Me![BUSCAR_COBERTURA] & "'"
Me.Bookmark = rs.Bookmark
COBERTURA1 = COBERTURA
VALOR_COBER1 = VALOR_COBER
End Sub

Private Sub BUSCAR_COBERTURA_KeyPress(Keycode As Integer)
If Keycode >= 0 And Keycode <= 255 Then
Keycode = 0
Beep
End If
End Sub
```

```

End If

End Sub

Private Sub Form_Open(Cancel As Integer)
Inicio
End Sub

Private Sub VALOR_COBER_KeyPress(Keycode As Integer)
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc(
"9")) Or Keycode = 45 Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub btnGuardar_Click()
If BUSCAR <> "" Then
BAN = 0
var = COBERTURA
pos = Form_MODIFICAR_COBERTURA.CurrentRecord
DoCmd.GoToRecord , , acGoTo, POSINI
If COBERTURA <> "" And VALOR_COBER <> 0 Then
    For I = 1 To BUSCAR_COBERTURA.ListCount
        If var = COBERTURA And pos <> Form_MODIFICAR_COBERTURA.CurrentRecord Then
            MsgBox "Cobertura existente", vbInformation, "SAVS"
            DoCmd.GoToRecord , , acGoTo, pos
            COBERTURA = COBERTURAl
            VALOR_COBER = VALOR_COBERl
            I = BUSCAR_COBERTURA.ListCount + 1
            BAN = 1
        End If
        If Form_MODIFICAR_COBERTURA.CurrentRecord <> POSINI + BUSCAR_COBERTURA.ListCount - 1 The
n
            DoCmd.GoToRecord , , acNext
        End If
    Next I
    Inicio
    If BAN = 0 Then
        MsgBox "Cobertura Modificada", vbInformation, "SAVS"
    End If
Else
    MsgBox "Ingrese los datos de la Cobertura", vbInformation, "SAVS"
End If
Else
    MsgBox "Ingrese el Código de la Póliza", vbInformation, "SAVS"
End If
End Sub

Private Sub btnEliminar_Click()
If BUSCAR <> "" Then
If BUSCAR_COBERTURA.ListCount > 1 Then
    respuesta = MsgBox("¿Está seguro de eliminar esta Cobertura?", vbYesNo, "SAVS")
    If respuesta = vbYes Then
        DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
        DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
        Inicio
    End If
Else
    MsgBox "Unica Cobertura, debe tener al menos una", vbInformation, "SAVS"
End If

```

```
Else
MsgBox "Ingrese el Código de la Póliza", vbInformation, "SAVS"
End If
End Sub
Private Sub btnSalir_Click()
If COBERTURA.Locked = True Then
    DoCmd.Close
Else
    MsgBox "Debe guardar antes de Salir", vbInformation, "SAVS"
End If
End Sub
```

Form_MODIFICAR_FORMA_PAGO - 1

Option Compare Database

Public DESCRIPCION1 As String

Public CANT_PAGOS1 As String

Private Sub btnDeshacer_Click()

On Error GoTo Err_btnDeshacer_Click

DoCmd.DoMenuItem acFormBar, acEditMenu, acUndo, , acMenuVer70

DoCmd.GoToRecord , , acNewRec

DESCRIPCION.Locked = True

CANT_PAGOS.Locked = True

BUSCAR_FORMA_PAGO = ""

Exit_btnDeshacer_Click:

Exit Sub

Err_btnDeshacer_Click:

MsgBox "No hay cambios que deshacer", vbInformation, "SAVS"

DoCmd.GoToRecord , , acNewRec

DESCRIPCION.Locked = True

CANT_PAGOS.Locked = True

BUSCAR_FORMA_PAGO = ""

Resume Exit_btnDeshacer_Click

End Sub

Private Sub btnGuardar_Click()

If BUSCAR_FORMA_PAGO <> "" Then

BAN = 0

var = DESCRIPCION

pos = Form_MODIFICAR_FORMA_PAGO.CurrentRecord

DoCmd.GoToRecord , , acLast

If DESCRIPCION <> "" And CANT_PAGOS <> 0 Then

HASTA = Form_MODIFICAR_FORMA_PAGO.CurrentRecord

For I = HASTA To 1 Step -1

If var = DESCRIPCION And pos <> Form_MODIFICAR_FORMA_PAGO.CurrentRecord Then

MsgBox "Forma de Pago existente", vbInformation, "SAVS"

DoCmd.GoToRecord , , acGoTo, pos

DESCRIPCION = DESCRIPCION1

CANT_PAGOS = CANT_PAGOS1

I = 0

BAN = 1

End If

If Form_MODIFICAR_FORMA_PAGO.CurrentRecord <> 1 Then

DoCmd.GoToRecord , , acPrevious

End If

Next I

DoCmd.GoToRecord , , acGoTo, pos

If BAN = 0 Then

MsgBox "Forma de Pago Modificada", vbInformation, "SAVS"

End If

Else

MsgBox "Ingrese los datos de la Forma de Pago", vbInformation, "SAVS"

End If

Else

MsgBox "Ingrese la Forma de Pago", vbInformation, "SAVS"

End If

DESCRIPCION.Locked = True

CANT_PAGOS.Locked = True

BUSCAR_FORMA_PAGO.Locked = False

End Sub

Private Sub btnSalir_Click()

If DESCRIPCION.Locked = True Then

```

        DoCmd.Close
Else
    MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"
End If

End Sub

Private Sub BUSCAR_FORMA_PAGO_KeyPress(Keycode As Integer)
If Keycode >= 0 And Keycode <= 255 Then
    Keycode = 0
    Beep
End If

End Sub

Private Sub CANT_PAGOS_KeyPress(Keycode As Integer)
If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 45
Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub DESCRIPCION_KeyPress(Keycode As Integer)
If Keycode = Asc("á") Or Keycode = Asc("é") Or Keycode = Asc("í") Or Keycode = Asc("ó") Or Keycode = Asc("ú") Or (Keycode >= 97 And Keycode <= 122) Or (Keycode >= 65 And Keycode <= 90) Or Keycode = 8 Or Keycode = 32 Or Keycode = 209 Or Keycode = 13 Or Keycode = 241 Then

Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub Form_Open(Cancel As Integer)
DESCRIPCION.Locked = True
CANT_PAGOS.Locked = True
DoCmd.GoToRecord , , acNewRec
End Sub

Private Sub BUSCAR_FORMA_PAGO_AfterUpdate()
' Buscar el registro que coincida con el control.
Dim rs As Object

Set rs = Me.Recordset.Clone
rs.FindFirst "[FORMA_PAGO] = " & Str(Me![BUSCAR_FORMA_PAGO])
Me.Bookmark = rs.Bookmark

DESCRIPCION1 = DESCRIPCION
CANT_PAGOS1 = CANT_PAGOS
DESCRIPCION.Locked = False
CANT_PAGOS.Locked = False
DoCmd.GoToControl "DESCRIPCION"
BUSCAR_FORMA_PAGO.Locked = True

End Sub

```

Option Compare Database

Private Sub btnDeshacer_Click()

On Error GoTo Err_btnDeshacer_Click

DoCmd.DoMenuItem acFormBar, acEditMenu, acUndo, , acMenuVer70

Inicio

Exit_btnDeshacer_Click:

Exit Sub

Err_btnDeshacer_Click:

MsgBox "No hay cambios que deshacer", vbInformation, "SAVS"

Inicio

Resume Exit_btnDeshacer_Click

End Sub

Private Sub btnGuardar_Click()

If ID_PROVEEDOR <> 0 And ID_POL <> "" And DESCRIPCION <> "" And VALOR_COSTO <> 0 And CANT_BENEFICI
C <> 0 And VALOR_VENTA <> 0 And PORC_COMISION <> 0 Then

DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70

Inicio

Else

MsgBox "Debe ingresar los datos de la Póliza", vbInformation, "SAVS"

End If

End Sub

Private Sub btnSalir_Click()

If ID_PROVEEDOR.Locked = False Then

MsgBox "Debe guardar antes de Salir", vbInformation, "SAVS"

Else

DoCmd.Close

End If

End Sub

Private Sub BUSCAR_AfterUpdate()

Set rs = Me.Recordset.Clone

rs.FindFirst "[ID_POL] = '" & Me![BUSCAR] & "'"

Me.Bookmark = rs.Bookmark

If ID_POL = BUSCAR Then

ID_PROVEEDOR.Locked = False

DESCRIPCION.Locked = False

PORC_COMISION.Locked = False

DoCmd.GoToControl "ID_PROVEEDOR"

BUSCAR.Locked = True

Else

BUSCAR = ""

DoCmd.GoToRecord , , acNewRec

MsgBox "Esta Póliza no existe", vbInformation, "SAVS"

End If

End Sub

Private Sub CANT_BENEFIC_KeyPress(Keycode As Integer)

If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13

Or Keycode = 28 Or Keycode = 9 Then

Else

Keycode = 0

Beep

End If

```
End Sub
Sub Inicio()
ID_PROVEEDOR.Locked = True
DESCRIPCION.Locked = True
VALOR_COSTO.Locked = True
CANT_BENEFIC.Locked = True
VALOR_VENTA.Locked = True
PORC_COMISION.Locked = True
BUSCAR = ""
BUSCAR.Locked = False
DoCmd.GoToRecord , , acNewRec

End Sub
Private Sub Form_Open(Cancel As Integer)
ID_POL.Visible = False
Inicio
End Sub

Private Sub ID_PROVEEDOR_KeyPress(Keycode As Integer)
If Keycode >= 0 And Keycode <= 255 Then
    Keycode = 0
    Beep
End If

End Sub

Private Sub PORC_COMISION_AfterUpdate()

If PORC_COMISION > 100 Then
    MsgBox "Porcentaje Inválido", vbInformation, "SAVS"
End If

End Sub

Private Sub PORC_COMISION_KeyPress(Keycode As Integer)
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub VALOR_COSTO_KeyPress(Keycode As Integer)
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub VALOR_VENTA_KeyPress(Keycode As Integer)
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If
```

End Sub

Option Compare Database

Public NOMBRE1 As String

Private Sub btnDeshacer_Click()

On Error GoTo Err_btnDeshacer_Click

DoCmd.DoMenuItem acFormBar, acEditMenu, acUndo, , acMenuVer70

DoCmd.GoToRecord , , acNewRec

NOMBRE.Locked = True

DIRECCION.Locked = True

FECHA_INGRESO.Locked = True

SALARIO.Locked = True

TELEFONO.Locked = True

CELULAR.Locked = True

BUSCAR_ID_PROMOTOR = ""

Exit_btnDeshacer_Click:

Exit Sub

Err_btnDeshacer_Click:

MsgBox "No hay cambios que deshacer", vbInformation, "SAVS"

DoCmd.GoToRecord , , acNewRec

NOMBRE.Locked = True

DIRECCION.Locked = True

FECHA_INGRESO.Locked = True

SALARIO.Locked = True

TELEFONO.Locked = True

CELULAR.Locked = True

BUSCAR_ID_PROMOTOR = ""

Resume Exit_btnDeshacer_Click

End Sub

Private Sub BUSCAR_ID_PROMOTOR_KeyPress(Keycode As Integer)

If Keycode >= 0 And Keycode <= 255 Then

Keycode = 0

Beep

End If

End Sub

Private Sub Form_Open(Cancel As Integer)

NOMBRE.Locked = True

DIRECCION.Locked = True

FECHA_INGRESO.Locked = True

SALARIO.Locked = True

TELEFONO.Locked = True

CELULAR.Locked = True

DoCmd.GoToRecord , , acNewRec

End Sub

Private Sub btnGuardar_Click()

If BUSCAR_ID_PROMOTOR <> "" Then

BAN = 0

var = NOMBRE

pos = Form_MODIFICAR_PROMOTOR.CurrentRecord

DoCmd.GoToRecord , , acLast

If NOMBRE <> "" And DIRECCION <> "" And FECHA_INGRESO <> "" And SALARIO <> 0 Then

HASTA = Form_MODIFICAR_PROMOTOR.CurrentRecord

For I = HASTA To 1 Step -1

If var = NOMBRE And pos <> Form_MODIFICAR_PROMOTOR.CurrentRecord Then

MsgBox "Promotor existente", vbInformation, "SAVS"

DoCmd.GoToRecord , , acGoTo, pos

NOMBRE = NOMBRE1

I = 0

```

        BAN = 1
    End If
    If Form_MODIFICAR_PROMOTOR.CurrentRecord <> 1 Then
        DoCmd.GoToRecord , , acPrevious
    End If
Next I
DoCmd.GoToRecord , , acGoTo, pos
If BAN = 0 Then
    MsgBox "Promotor Modificado", vbInformation, "SAVS"
End If
Else
    MsgBox "Ingrese los datos del Promotor", vbInformation, "SAVS"
End If
NOMBRE.Locked = True
DIRECCION.Locked = True
FECHA_INGRESO.Locked = True
SALARIO.Locked = True
TELEFONO.Locked = True
CELULAR.Locked = True
BUSCAR_ID_PROMOTOR.Locked = False
Else
    MsgBox "Ingrese los datos del Promotor", vbInformation, "SAVS"
End If
End Sub
Private Sub btnSalir_Click()
    If NOMBRE.Locked = True Then
        DoCmd.Close
    Else
        MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"
    End If
End Sub

Private Sub NOMBRE_KeyPress(Keycode As Integer)
    If Keycode = Asc("á") Or Keycode = Asc("é") Or Keycode = Asc("í") Or Keycode = Asc("ó") Or Keycode = Asc("ú") Or (Keycode >= 97 And Keycode <= 122) Or (Keycode >= 65 And Keycode <= 90) Or Keycode = 8 Or Keycode = 32 Or Keycode = 209 Or Keycode = 13 Or Keycode = 241 Then

    Else
        Keycode = 0
        Beep
    End If
End Sub

Private Sub SALARIO_KeyPress(Keycode As Integer)
    If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
    Else
        Keycode = 0
        Beep
    End If
End Sub
Private Sub BUSCAR_ID_PROMOTOR_AfterUpdate()
    ' Buscar el registro que coincida con el control.
    Dim rs As Object

    Set rs = Me.Recordset.Clone
    rs.FindFirst "[ID_PROMOTOR] = " & Str(Me![BUSCAR_ID_PROMOTOR])
    Me.Bookmark = rs.Bookmark

```

```
NOMBRE1 = NOMBRE  
NOMBRE.Locked = False  
DIRECCION.Locked = False  
FECHA_INGRESO.Locked = False  
SALARIO.Locked = False  
TELEFONO.Locked = False  
CELULAR.Locked = False  
DoCmd.GoToControl "NOMBRE"  
BUSCAR_ID_PROMOTOR.Locked = True
```

End Sub

Option Compare Database
Public PROVEEDOR1 As String

```
Private Sub btnDeshacer_Click()  
On Error GoTo Err_btnDeshacer_Click  
    DoCmd.DoMenuItem acFormBar, acEditMenu, acUndo, , acMenuVer70  
    DoCmd.GoToRecord , , acNewRec  
    PROVEEDOR.Locked = True  
    DIRECCION.Locked = True  
    TELEFONO.Locked = True  
    CONTACTO.Locked = True  
    EXTENCION.Locked = True  
    BUSCAR_PROVEEDOR = ""  
Exit_btnDeshacer_Click:
```

```
    Exit Sub  
Err_btnDeshacer_Click:  
    MsgBox "No hay cambios que deshacer", vbInformation, "SAVS"  
    DoCmd.GoToRecord , , acNewRec  
    PROVEEDOR.Locked = True  
    DIRECCION.Locked = True  
    TELEFONO.Locked = True  
    CONTACTO.Locked = True  
    EXTENCION.Locked = True  
    BUSCAR_PROVEEDOR = ""  
Resume Exit_btnDeshacer_Click
```

End Sub

```
Private Sub BUSCAR_PROVEEDOR_KeyPress(Keycode As Integer)  
If Keycode >= 0 And Keycode <= 255 Then  
    Keycode = 0  
    Beep  
End If
```

End Sub

```
Private Sub EXTENCION_KeyPress(Keycode As Integer)  
If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 45  
Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then  
Else  
    Keycode = 0  
    Beep  
End If
```

End Sub

```
Private Sub btnGuardar_Click()  
If BUSCAR_PROVEEDOR <> "" Then  
    BAN = 0  
    var = PROVEEDOR  
    pos = Form_MODIFICAR_PROVEEDOR.CurrentRecord  
    DoCmd.GoToRecord , , acLast  
    If PROVEEDOR <> "" And DIRECCION <> "" And TELEFONO <> "" Then  
        HASTA = Form_MODIFICAR_PROVEEDOR.CurrentRecord  
        For I = HASTA To 1 Step -1  
            If var = PROVEEDOR And pos <> Form_MODIFICAR_PROVEEDOR.CurrentRecord Then  
                MsgBox "Proveedor existente", vbInformation, "SAVS"  
                DoCmd.GoToRecord , , acGoTo, pos  
                PROVEEDOR = PROVEEDOR1  
                I = 0  
                BAN = 1
```

```
End If
If Form_MODIFICAR_PROVEEDOR.CurrentRecord <> 1 Then
    DoCmd.GoToRecord , , acPrevious
End If
Next I
DoCmd.GoToRecord , , acGoTo, pos
If BAN = 0 Then
    MsgBox "Proveedor Modificado", vbInformation, "SAVS"
End If
Else
    MsgBox "Ingrese los datos del Proveedor", vbInformation, "SAVS"
End If
Else
    MsgBox "Ingrese los datos del Proveedor", vbInformation, "SAVS"
End If
PROVEEDOR.Locked = True
DIRECCION.Locked = True
TELEFONO.Locked = True
CONTACTO.Locked = True
EXTENCION.Locked = True
BUSCAR_PROVEEDOR.Locked = False

End Sub
Private Sub btnSalir_Click()

If PROVEEDOR.Locked = True Then
    DoCmd.Close
Else
    MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"
End If

End Sub

Private Sub Form_Open(Cancel As Integer)
PROVEEDOR.Locked = True
DIRECCION.Locked = True
TELEFONO.Locked = True
CONTACTO.Locked = True
EXTENCION.Locked = True

DoCmd.GoToRecord , , acNewRec
End Sub
Private Sub BUSCAR_PROVEEDOR_AfterUpdate()
' Buscar el registro que coincida con el control.
Dim rs As Object

Set rs = Me.Recordset.Clone
rs.FindFirst "[ID_PROVEEDOR] = " & Str(Me![BUSCAR_PROVEEDOR])
Me.Bookmark = rs.Bookmark

PROVEEDOR1 = PROVEEDOR
PROVEEDOR.Locked = False
DIRECCION.Locked = False
TELEFONO.Locked = False
CONTACTO.Locked = False
EXTENCION.Locked = False
DoCmd.GoToControl "PROVEEDOR"
BUSCAR_PROVEEDOR.Locked = True
End Sub
```

Form_MODIFICAR_TASA - 1

Option Compare Database

```
Private Sub Form_Open(Cancel As Integer)
```

```
FECHA.Locked = True
```

```
DoCmd.GoToControl "TASA"
```

```
DoCmd.GoToRecord , , acLast
```

```
End Sub
```

```
Private Sub btnGuardar_Click()
```

```
If TASA <> "" Then
```

```
    If TASA > 0 Then
```

```
        DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
```

```
        DoCmd.Close
```

```
    Else
```

```
        MsgBox "Tasa de Cambio Inválida", vbInformation, "SAVS"
```

```
        TASA = 0
```

```
    End If
```

```
Else
```

```
    MsgBox "Ingrese la Tasa de Cambio de Hoy", vbInformation, "SAVS"
```

```
End If
```

```
End Sub
```

```
Private Sub TASA_KeyPress(Keycode As Integer)
```

```
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
```

```
Else
```

```
    Keycode = 0
```

```
    Beep
```

```
End If
```

```
End Sub
```

Form_MODIFICAR_TIPO_ID - 1

Option Compare Database

Public DESCRIPCION1 As String

Private Sub btnDeshacer_Click()

On Error GoTo Err_btnDeshacer_Click

DoCmd.DoMenuItem acFormBar, acEditMenu, acUndo, , acMenuVer70

DoCmd.GoToRecord , , acNewRec

DESCRIPCION.Locked = True

BUSCAR_TIPO_ID = ""

Exit_btnDeshacer_Click:

Exit Sub

Err_btnDeshacer_Click:

MsgBox "No hay cambios que deshacer", vbInformation, "SAVS"

DoCmd.GoToRecord , , acNewRec

DESCRIPCION.Locked = True

BUSCAR_TIPO_ID = ""

Resume Exit_btnDeshacer_Click

End Sub

Private Sub btnGuardar_Click()

If BUSCAR_TIPO_ID <> "" Then

BAN = 0

var = DESCRIPCION

pos = Form_MODIFICAR_TIPO_ID.CurrentRecord

DoCmd.GoToRecord , , acLast

If DESCRIPCION <> "" Then

HASTA = Form_MODIFICAR_TIPO_ID.CurrentRecord

For I = HASTA To 1 Step -1

If var = DESCRIPCION And pos <> Form_MODIFICAR_TIPO_ID.CurrentRecord Then

MsgBox "Tipo de Identificación existente", vbInformation, "SAVS"

DoCmd.GoToRecord , , acGoTo, pos

DESCRIPCION = DESCRIPCION1

I = 0

BAN = 1

End If

If Form_MODIFICAR_TIPO_ID.CurrentRecord <> 1 Then

DoCmd.GoToRecord , , acPrevious

End If

Next I

DoCmd.GoToRecord , , acGoTo, pos

If BAN = 0 Then

MsgBox "Tipo de Identificación Modificada", vbInformation, "SAVS"

End If

Else

MsgBox "Ingrese el Tipo de Identificación", vbInformation, "SAVS"

End If

Else

MsgBox "Ingrese el Tipo de Identificación", vbInformation, "SAVS"

End If

DESCRIPCION.Locked = True

BUSCAR_TIPO_ID.Locked = False

End Sub

Private Sub btnSalir_Click()

If DESCRIPCION.Locked = True Then

DoCmd.Close

Else

MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"

End If

End Sub

```
Private Sub BUSCAR_TIPO_ID_AfterUpdate()  
    ' Buscar el registro que coincida con el control.  
    Dim rs As Object  
  
    Set rs = Me.Recordset.Clone  
    rs.FindFirst "[TIPO_ID] = " & Str(Me![BUSCAR_TIPO_ID])  
    Me.Bookmark = rs.Bookmark  
    DESCRIPCION1 = DESCRIPCION  
    DESCRIPCION.Locked = False  
    DoCmd.GoToControl "DESCRIPCION"  
    BUSCAR_TIPO_ID.Locked = True
```

End Sub

```
Private Sub BUSCAR_TIPO_ID_KeyPress(Keycode As Integer)  
    If Keycode >= 0 And Keycode <= 255 Then  
        Keycode = 0  
        Beep  
    End If  
End Sub
```

```
Private Sub Form_Open(Cancel As Integer)  
    DESCRIPCION.Locked = True  
    DoCmd.GoToRecord , , acNewRec  
End Sub
```

Form_MODIFICAR_TIPO_PAGO - 1

Option Compare Database

Public TIPO_PAGO1 As String

Private Sub btnDeshacer_Click()

On Error GoTo Err_btnDeshacer_Click

DoCmd.DoMenuItem acFormBar, acEditMenu, acUndo, , acMenuVer70

DoCmd.GoToRecord , , acNewRec

TIPO_PAGO.Locked = True

BUSCAR_TIPO_PAGO = ""

Exit_btnDeshacer_Click:

Exit Sub

Err_btnDeshacer_Click:

MsgBox "No hay cambios que deshacer", vbInformation, "SAVS"

DoCmd.GoToRecord , , acNewRec

TIPO_PAGO.Locked = True

BUSCAR_TIPO_PAGO = ""

Resume Exit_btnDeshacer_Click

End Sub

Private Sub btnGuardar_Click()

If BUSCAR_TIPO_PAGO <> "" Then

BAN = 0

var = TIPO_PAGO

pos = Form_MODIFICAR_TIPO_PAGO.CurrentRecord

DoCmd.GoToRecord , , acLast

If TIPO_PAGO <> "" Then

HASTA = Form_MODIFICAR_TIPO_PAGO.CurrentRecord

For I = HASTA To 1 Step -1

If var = TIPO_PAGO And pos <> Form_MODIFICAR_TIPO_PAGO.CurrentRecord Then

MsgBox "Tipo de Pago existente", vbInformation, "SAVS"

DoCmd.GoToRecord , , acGoTo, pos

TIPO_PAGO = TIPO_PAGO1

I = 0

BAN = 1

End If

If Form_MODIFICAR_TIPO_PAGO.CurrentRecord <> 1 Then

DoCmd.GoToRecord , , acPrevious

End If

Next I

DoCmd.GoToRecord , , acGoTo, pos

If BAN = 0 Then

MsgBox "Tipo de Pago Modificado", vbInformation, "SAVS"

End If

Else

MsgBox "Ingrese el Tipo de Pago", vbInformation, "SAVS"

End If

TIPO_PAGO.Locked = True

BUSCAR_TIPO_PAGO.Locked = False

Else

MsgBox "Ingrese el Tipo de Pago", vbInformation, "SAVS"

End If

End Sub

Private Sub btnSalir_Click()

If TIPO_PAGO.Locked = True Then

DoCmd.Close

Else

MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"

End If

End Sub

```
Private Sub BUSCAR_TIPO_PAGO_AfterUpdate()  
    ' Buscar el registro que coincida con el control.  
    Dim rs As Object  
  
    Set rs = Me.Recordset.Clone  
    rs.FindFirst "[ID_TIPO_PAGO] = " & Str(Me![BUSCAR_TIPO_PAGO])  
    Me.Bookmark = rs.Bookmark  
  
    TIPO_PAGO1 = TIPO_PAGO  
    TIPO_PAGO.Locked = False  
    DoCmd.GoToControl "TIPO_PAGO"  
    BUSCAR_TIPO_PAGO.Locked = True  
  
End Sub  
  
Private Sub BUSCAR_TIPO_PAGO_KeyPress(Keycode As Integer)  
    If Keycode >= 0 And Keycode <= 255 Then  
        Keycode = 0  
        Beep  
    End If  
  
End Sub  
  
Private Sub Form_Open(Cancel As Integer)  
    TIPO_PAGO.Locked = True  
    DoCmd.GoToRecord , , acNewRec  
End Sub
```

Option Compare Database

Private Sub btnDeshacer_Click()

On Error GoTo Err_btnDeshacer_Click

DoCmd.DoMenuItem acFormBar, acEditMenu, acUndo, , acMenuVer70

DoCmd.GoToRecord , , acNewRec

ID_VENTA.Visible = False

ID_CLIENTE.Locked = True

ID_POL.Locked = True

ID_PROMOTOR.Locked = True

FECHA_ADQUI.Locked = True

FORMA_PAGO.Locked = True

BUSCAR = ""

Exit_btnDeshacer_Click:

Exit Sub

Err_btnDeshacer_Click:

MsgBox "No hay cambios que deshacer", vbInformation, "SAVS"

DoCmd.GoToRecord , , acNewRec

ID_VENTA.Visible = False

ID_CLIENTE.Locked = True

ID_POL.Locked = True

ID_PROMOTOR.Locked = True

FECHA_ADQUI.Locked = True

FORMA_PAGO.Locked = True

BUSCAR = ""

Resume Exit_btnDeshacer_Click

End Sub

Private Sub btnGuardar_Click()

If BUSCAR <> "" Then

DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70

ID_PROMOTOR.Locked = True

FORMA_PAGO.Locked = True

DoCmd.GoToRecord , , acNewRec

BUSCAR = ""

Else

MsgBox "Ingrese el Número de la Venta a Modificar", vbInformation, "SAVS"

End If

End Sub

Private Sub btnSalir_Click()

If ID_PROMOTOR.Locked = True Then

DoCmd.Close

Else

MsgBox "Debe guardar antes de Salir", vbInformation, "SAVS"

End If

End Sub

Private Sub BUSCAR_AfterUpdate()

Dim rs As Object

Set rs = Me.Recordset.Clone

rs.FindFirst "[ID_VENTA] = " & Str(Me![BUSCAR])

Me.Bookmark = rs.Bookmark

If ID_VENTA = BUSCAR Then

ID_PROMOTOR.Locked = False

FORMA_PAGO.Locked = False

Else

DoCmd.GoToRecord , , acNewRec

Form_MODIFICAR_VENTA - 2

```
MsgBox "Esta Venta no existe", vbInformation, "SAVS"  
BUSCAR = ""
```

End If

End Sub

```
Private Sub Form_Open(Cancel As Integer)  
DoCmd.GoToRecord , , acNewRec  
ID_VENTA.Visible = False  
ID_CLIENTE.Locked = True  
ID_POL.Locked = True  
ID_PROMOTOR.Locked = True  
FECHA_ADQUI.Locked = True  
FORMA_PAGO.Locked = True
```

End Sub

```
Private Sub FORMA_PAGO_KeyPress(Keycode As Integer)  
If Keycode >= 0 And Keycode <= 255 Then  
Keycode = 0  
Beep  
End If
```

End Sub

```
Private Sub ID_POL_AfterUpdate()  
DoCmd.Requery "FORMA_PAGO"  
End Sub
```

```
Private Sub ID_PROMOTOR_KeyPress(Keycode As Integer)  
If Keycode >= 0 And Keycode <= 255 Then  
Keycode = 0  
Beep  
End If
```

End Sub

```
Private Sub btnEliminar_Click()  
If ID_PROMOTOR.Locked = False Then  
respuesta = MsgBox("¿Está seguro de Eliminar esta Venta?", vbYesNo, "SAVS")  
If respuesta = vbYes Then  
DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70  
DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70  
End If  
Else  
MsgBox "No existe una Venta activa", vbInformation, "SAVS"  
End If
```

End Sub

Form_PAGO_COMISION - 1

Option Compare Database

Private Sub btnSalir_Click()

 DoCmd.Close

End Sub

Private Sub btnEjecutar_Click()

If FECHA1 <> "" And FECHA2 <> "" Then

 DoCmd.OpenReport "COMISIONES", acViewPreview

Else

 MsgBox "Ingrese los datos necesarios", vbInformation, "SAVS"

End If

End Sub

Form_PARAMETRO_VENTAS - 1

Option Compare Database

Private Sub BUSCAR_AfterUpdate()

DoCmd.OpenForm "BUSCAR_VENTAS", acNormal

End Sub

Private Sub BUSCAR_KeyPress(Keycode As Integer)

If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 45

Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then

Else

 Keycode = 0

 Beep

End If

End Sub

Option Compare Database

```

Private Sub btnGuardar_Click()
ID_POL = "0"
BAN = 0
DoCmd.GoToRecord , , acLast
If ID_PROVEEDOR <> 0 And ID_POL <> "" And DESCRIPCION <> "" And VALOR_COSTO <> 0 And CANT_BENEFI
C <> 0 And VALOR_VENTA <> 0 And PORC_COMISION <> 0 Then
    var = RAM
    HASTA = Form_POLIZA_COMPRA.CurrentRecord
    If HASTA = 1 Then
        Guarda_Abre
    Else
        For I = HASTA - 1 To 1 Step -1
            DoCmd.GoToRecord , , acPrevious
            If var = ID_POL Then
                MsgBox "Póliza existente", vbInformation, "SAVS"
                DoCmd.GoToRecord , , acLast
                DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
                DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
                DoCmd.GoToRecord , , acNewRec
                RAM = ""
                I = 0
                BAN = 1
            End If
        Next I
        If BAN = 0 Then
            DoCmd.GoToRecord , , acGoTo, HASTA
            Guarda_Abre
        End If
    End If
Else
    MsgBox "Ingrese los datos necesarios de la Póliza", vbInformation, "SAVS"
End If

End Sub

Sub Guarda_Abre()
Dim stLinkCriteria As String
ID_POL = RAM
DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
DoCmd.GoToRecord , , acLast
ID_PROVEEDOR.Locked = True
ID_POL.Locked = True
DESCRIPCION.Locked = True
VALOR_COSTO.Locked = True
CANT_BENEFIC.Locked = True
VALOR_VENTA.Locked = True
PORC_COMISION.Locked = True
DoCmd.GoToControl "RAM"
btnGuardar.Enabled = False
btnSalir.Enabled = False
stLinkCriteria = "[ID_POL]=" & "" & Me![ID_POL] & ""
DoCmd.OpenForm "COBERTURAS", , , stLinkCriteria
Form_COBERTURAS.ID_POL = Form_POLIZA_COMPRA.ID_POL

End Sub

Private Sub btnSalir_Click()

If ID_PROVEEDOR <> 0 And ID_POL <> "" And DESCRIPCION <> "" And VALOR_COSTO <> 0 And CANT_BENEFI
C <> 0 And VALOR_VENTA <> 0 And PORC_COMISION <> 0 Then
    MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"

```

```

Else
    If ID_PROVEEDOR <> 0 Or ID_POL <> "" Or DESCRIPCION <> "" Or VALOR_COSTO <> 0 Or CANT_BENEFI
C <> 0 Or VALOR_VENTA <> 0 Or PORC_COMISION <> 0 Then
        respuesta = MsgBox("Datos no guardados, ¿Desea Salir?", vbYesNo, "SAVS")
        If respuesta = vbYes Then
            ID_POL = "0"
            DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
            DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
            DoCmd.Close
        End If
    Else
        DoCmd.Close
    End If
End If

End Sub

Private Sub CANT_BENEFIC_KeyPress(Keycode As Integer)
If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13
Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub Form_Open(Cancel As Integer)
RAM = ""
ID_POL.Visible = False
DoCmd.GoToRecord , , acNewRec
End Sub

Private Sub ID_PROVEEDOR_KeyPress(Keycode As Integer)
If Keycode >= 0 And Keycode <= 255 Then
    Keycode = 0
    Beep
End If

End Sub

Private Sub PORC_COMISION_AfterUpdate()
If PORC_COMISION > 100 Then
    MsgBox "Porcentaje Inválido", vbInformation, "SAVS"
End If

End Sub

Private Sub PORC_COMISION_KeyPress(Keycode As Integer)
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc(
"9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub VALOR_COSTO_KeyPress(Keycode As Integer)
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc(

```

```
"9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
```

```
Else
```

```
    Keycode = 0
```

```
    Beep
```

```
End If
```

```
End Sub
```

```
Private Sub VALOR_VENTA_KeyPress(Keycode As Integer)
```

```
If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
```

```
Else
```

```
    Keycode = 0
```

```
    Beep
```

```
End If
```

```
End Sub
```

Form_PROMOTORES - 1

Option Compare Database

Private Sub Form_Open(Cancel As Integer)

DoCmd.GoToRecord , , acNewRec

End Sub

Private Sub btnGuardar_Click()

BAN = 0

DoCmd.GoToRecord , , acLast

If NOMBRE <> "" And DIRECCION <> "" And FECHA_INGRESO <> "" And SALARIO <> 0 Then

var = NOMBRE

HASTA = Form_PROMOTORES.CurrentRecord

If HASTA = 1 Then

DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70

DoCmd.GoToRecord , , acNewRec

Else

For I = HASTA - 1 To 1 Step -1

DoCmd.GoToRecord , , acPrevious

If var = NOMBRE Then

MsgBox "Promotor existente", vbInformation, "SAVS"

DoCmd.GoToRecord , , acLast

I = 0

BAN = 1

End If

Next I

If BAN = 0 Then

DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70

DoCmd.GoToRecord , , acNewRec

End If

End If

Else

MsgBox "Ingrese los datos del Promotor", vbInformation, "SAVS"

End If

End Sub

Private Sub btnSalir_Click()

If ID_PROMOTOR <> "" Then

If NOMBRE <> "" And DIRECCION <> "" And FECHA_INGRESO <> "" And SALARIO <> 0 Then

MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"

Else

respuesta = MsgBox("Datos no guardados, ¿Desea Salir?", vbYesNo, "SAVS")

If respuesta = vbYes Then

DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70

DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70

DoCmd.Close

End If

End If

Else

DoCmd.Close

End If

End Sub

Private Sub NOMBRE_KeyPress(Keycode As Integer)

If Keycode = Asc("á") Or Keycode = Asc("é") Or Keycode = Asc("í") Or Keycode = Asc("ó") Or Keycode = Asc("ú") Or (Keycode >= 97 And Keycode <= 122) Or (Keycode >= 65 And Keycode <= 90) Or Keycode = 8 Or Keycode = 32 Or Keycode = 209 Or Keycode = 13 Or Keycode = 241 Then

Else

Keycode = 0

Beep

End If

End Sub

Private Sub SALARIO_KeyPress(Keycode As Integer)

If Keycode = Asc(".") Or Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then

Else

 Keycode = 0

 Beep

End If

End Sub

Option Compare Database

```

Private Sub EXTENCION_KeyPress(Keycode As Integer)
If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 45
Or Keycode = 13 Or Keycode = 28 Or Keycode = 9 Then
Else
    Keycode = 0
    Beep
End If

End Sub

Private Sub btnGuardar_Click()

BAN = 0
DoCmd.GoToRecord , , acLast
If PROVEEDOR <> "" And DIRECCION <> "" And TELEFONO <> "" Then
    var = PROVEEDOR
    HASTA = Form_PROVEEDORES.CurrentRecord
    If HASTA = 1 Then
        DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
        DoCmd.GoToRecord , , acNewRec
    Else
        For I = HASTA - 1 To 1 Step -1
            DoCmd.GoToRecord , , acPrevious
            If var = PROVEEDOR Then
                MsgBox "Proveedor existente", vbInformation, "SAVS"
                DoCmd.GoToRecord , , acLast
                I = 0
                BAN = 1
            End If
        Next I
        If BAN = 0 Then
            DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
            DoCmd.GoToRecord , , acNewRec
        End If
    End If
Else
    MsgBox "Ingrese los datos necesarios del Proveedor", vbInformation, "SAVS"
End If

End Sub

Private Sub btnSalir_Click()

If ID_PROVEEDOR <> "" Then
    If PROVEEDOR <> "" And DIRECCION <> "" And TELEFONO <> "" Then
        MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"
    Else
        respuesta = MsgBox("Datos no guardados, ¿Desea Salir?", vbYesNo, "SAVS")
        If respuesta = vbYes Then
            DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
            DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
            DoCmd.Close
        End If
    End If
Else
    DoCmd.Close
End If

End Sub

```

Form_PROVEEDORES - 2

```
Private Sub Form_Open(Cancel As Integer)
DoCmd.GoToRecord , , acNewRec
End Sub
```

Form_REPORTES_VENTAS - 1

Option Compare Database

Private Sub btnEjecutar_Click()

If FECHA1 <> "" And FECHA2 <> "" Then

 If OPCION = 1 Then

 DoCmd.OpenReport "VENTAS_PROMOTOR", acViewPreview

 Else

 If OPCION = 2 Then

 DoCmd.OpenReport "VENTAS_POLIZA", acViewPreview

 Else

 If OPCION = 3 Then

 DoCmd.OpenReport "VENTAS_VENCERSE", acViewPreview

 Else

 DoCmd.OpenReport "VENTAS_FECHA", acViewPreview

 End If

 End If

 End If

Else

 MsgBox "Ingrese los parámetros necesarios", vbInformation, "SAVS"

End If

End Sub

Private Sub btnSalir_Click()

 DoCmd.Close

End Sub

Form_RESPALDO - 1

Option Compare Database

Private Sub Comando4_Click()

On Error GoTo Err_Comando4_Click

Dim stAppName As String

stAppName = "msaccess.exe c:\Respaldo.mdb"

Call Shell(stAppName, 1)

Exit_Comando4_Click:

Exit Sub

Err_Comando4_Click:

MsgBox Err.Description

Resume Exit_Comando4_Click

End Sub

Option Compare Database

Sub Inicio()

```
TASA_CAMBIO.Visible = False
DoCmd.GoToControl "BUSCAR"
ID_RECIBO.Visible = False
FECHA_CANCEL.Visible = False
ESTADO.Visible = False
VALOR_TOTAL_CORDOBAS.Visible = False
CORDOBAS.Locked = True
DOLARES.Locked = True
TASA.Locked = True
NOMBRE.Locked = True
VALOR_RECIBO.Locked = True
TOTAL_CORDOBAS.Locked = True
LETRAS.Locked = True
TIPO_PAGO.Locked = True
ID_BANCO.Locked = True
REFERENCIA.Locked = True
BUSCAR.Locked = False
TOTAL_CORDOBAS = ""
TASA = ""
DoCmd.GoToRecord , , acNewRec
End Sub
```

Private Sub btnSalir_Click()

DoCmd.Close

End Sub

Private Sub BUSCAR_AfterUpdate()

Dim rs As Object

Set rs = Me.Recordset.Clone

rs.FindFirst "[ID_RECIBO] = " & Str(Me![BUSCAR])

Me.Bookmark = rs.Bookmark

If ID_RECIBO = BUSCAR Then

 If ESTADO = "CANCELADO" Then

 BUSCAR.Locked = True

 Else

 Inicio

 MsgBox "Está Emitido el Recibo N° " & BUSCAR, vbInformation, "SAVS"

 BUSCAR = ""

 DoCmd.GoToControl "BUSCAR"

 End If

Else

 Inicio

 MsgBox "No existe el Recibo N° " & BUSCAR, vbInformation, "SAVS"

 BUSCAR = ""

 DoCmd.GoToControl "BUSCAR"

End If

End Sub

Private Sub Form_Open(Cancel As Integer)

Inicio

End Sub

Private Sub btnRevertir_Click()

respuesta = MsgBox("¿Desea Revertir el Recibo?", vbYesNo, "SAVS")

If respuesta = vbYes Then

 ESTADO = "EMITIDO"

 FECHA_CANCEL = ""

```
DOLARES = ""  
CORDOBAS = ""  
LETRAS = ""  
REFERENCIA = ""  
ID_BANCO = ""  
TIPO_PAGO = ""  
VALOR_TOTAL_CORDOBAS = ""  
BUSCAR = ""  
Inicio
```

```
Else  
End If  
End Sub
```

Form_TASA - 1

Option Compare Database

```
Private Sub Form_Open(Cancel As Integer)
```

```
FECHA.Locked = True
```

```
DoCmd.GoToControl "TASA"
```

```
DoCmd.GoToRecord , , acLast
```

```
If FECHA <> Date Then
```

```
    DoCmd.GoToRecord , , acNewRec
```

```
Else
```

```
    DoCmd.Close
```

```
End If
```

```
End Sub
```

```
Private Sub btnGuardar_Click()
```

```
If TASA <> "" Then
```

```
    If TASA > 0 Then
```

```
        DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
```

```
        DoCmd.Close
```

```
    Else
```

```
        MsgBox "Tasa de Cambio Inválida", vbInformation, "SAVS"
```

```
TASA = 0
```

```
    End If
```

```
Else
```

```
    MsgBox "Ingrese la Tasa de Cambio de Hoy", vbInformation, "SAVS"
```

```
End If
```

```
End Sub
```

```
Private Sub TASA_KeyPress(Keycode As Integer)
```

```
If Keycode = 8 Or Keycode = 27 Or (Keycode >= Asc("0") And Keycode <= Asc("9")) Or Keycode = 13
```

```
Or Keycode = 28 Or Keycode = 9 Then
```

```
Else
```

```
    Keycode = 0
```

```
    Beep
```

```
End If
```

```
End Sub
```

Form_TIPO_ID - 1

Option Compare Database

```
Private Sub btnGuardar_Click()  
BAN = 0  
DoCmd.GoToRecord , , acLast  
If DESCRIPCION <> "" Then  
    var = DESCRIPCION  
    HASTA = Form_TIPO_ID.CurrentRecord  
    If HASTA = 1 Then  
        DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70  
        DoCmd.GoToRecord , , acNewRec  
    Else  
        For I = HASTA - 1 To 1 Step -1  
            DoCmd.GoToRecord , , acPrevious  
            If var = DESCRIPCION Then  
                MsgBox "Tipo de Identificación existente", vbInformation, "SAVS"  
                DoCmd.GoToRecord , , acLast  
                I = 0  
                BAN = 1  
            End If  
        Next I  
        If BAN = 0 Then  
            DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70  
            DoCmd.GoToRecord , , acNewRec  
        End If  
    End If  
Else  
    MsgBox "Ingrese el tipo de Identificación", vbInformation, "SAVS"  
End If  
  
End Sub  
Private Sub btnSalir_Click()  
  
If TIPO_ID <> "" Then  
    If DESCRIPCION <> "" Then  
        MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"  
    Else  
        DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70  
        DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70  
        DoCmd.Close  
    End If  
Else  
    DoCmd.Close  
End If  
  
End Sub  
  
Private Sub Form_Open(Cancel As Integer)  
DoCmd.GoToRecord , , acNewRec  
End Sub
```

Form_TIPO_PAGO - 1

Option Compare Database

```
Private Sub btnGuardar_Click()
BAN = 0
DoCmd.GoToRecord , , acLast
If TIPO_PAGO <> "" Then
    var = TIPO_PAGO
    HASTA = Form_TIPO_PAGO.CurrentRecord
    If HASTA = 1 Then
        DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
        DoCmd.GoToRecord , , acNewRec
    Else
        For I = HASTA - 1 To 1 Step -1
            DoCmd.GoToRecord , , acPrevious
            If var = TIPO_PAGO Then
                MsgBox "Modalidad de Pago existente", vbInformation, "SAVS"
                DoCmd.GoToRecord , , acLast
                I = 0
                BAN = 1
            End If
        Next I
        If BAN = 0 Then
            DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70
            DoCmd.GoToRecord , , acNewRec
        End If
    End If
Else
    MsgBox "Ingrese la Modalidad de pago", vbInformation, "SAVS"
End If

End Sub

Private Sub btnSalir_Click()
If ID_TIPO_PAGO <> "" Then
    If TIPO_PAGO <> "" Then
        MsgBox "Debe guardar antes de salir", vbInformation, "SAVS"
    Else
        DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70
        DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70
        DoCmd.Close
    End If
Else
    DoCmd.Close
End If

End Sub

Private Sub Form_Open(Cancel As Integer)
DoCmd.GoToRecord , , acNewRec
End Sub
```

Form_ventanadecopia - 1

Option Compare Database

Private Sub Form_Load()

Quit

End Sub

Private Sub Form_Unload(Cancel As Integer)

Dim FrmVentaActiva As Form

Set FrmVentaActiva = Me

ShellExecute FrmVentaActiva.hwnd, "open", CurrentProject.Path & "\Respaldar.mdb", "", "", SW_SH

OW

End Sub

Form_VENTAS - 1

Option Compare Database

Private Sub btnGuardar_Click()

If ID_CLIENTE <> 0 And ID_POL <> 0 And ID_PROMOTOR <> 0 And FORMA_PAGO <> 0 And FECHA_ADQUI <> "

Then
 VENCIMIENTO = DateSerial(Year([FECHA_ADQUI]) + 1, Month([FECHA_ADQUI]), Day([FECHA_ADQUI]))
- 1

 Guarda_Abre

Else

 MsgBox "Complete los datos de la Venta", vbInformation, "SAVS"

End If

End Sub

Sub Guarda_Abre()

DoCmd.DoMenuItem acFormBar, acRecordsMenu, acSaveRecord, , acMenuVer70

DoCmd.OpenQuery "RECIBO", acViewNormal, acEdit

Dim stLinkCriteria As String

FECHA_ADQUI.Locked = True

ID_CLIENTE.Locked = True

ID_POL.Locked = True

ID_PROMOTOR.Locked = True

FORMA_PAGO.Locked = True

DoCmd.GoToControl "ID_VENTA"

btnGuardar.Enabled = False

btnSalir.Enabled = False

stLinkCriteria = "[ID_VENTA]=" & Me![ID_VENTA]

DoCmd.OpenForm "BENEFICIARIOS", , , stLinkCriteria

Form_BENEFICIARIOS.ID_VENTA = Form_VENTAS.ID_VENTA

End Sub

Private Sub btnSalir_Click()

If ID_VENTA <> "" Then

 If ID_CLIENTE <> 0 And ID_POL <> 0 And ID_PROMOTOR <> 0 And FORMA_PAGO <> 0 And FECHA_ADQUI <> "" Then

 MsgBox "Debe guardar antes de Salir", vbInformation, "SAVS"

 Else

 respuesta = MsgBox("Datos no guardados, ¿Desea Salir?", vbYesNo, "SAVS")

 If respuesta = vbYes Then

 DoCmd.DoMenuItem acFormBar, acEditMenu, 8, , acMenuVer70

 DoCmd.DoMenuItem acFormBar, acEditMenu, 6, , acMenuVer70

 DoCmd.Close

 End If

 End If

Else

 DoCmd.Close

End If

End Sub

Private Sub Form_Open(Cancel As Integer)

DoCmd.GoToRecord , , acNewRec

End Sub

Private Sub FORMA_PAGO_KeyPress(Keycode As Integer)

If Keycode >= 0 And Keycode <= 255 Then

 Keycode = 0

 Beep

End If

End Sub

Private Sub ID_CLIENTE_AfterUpdate()

If ID_CLIENTE <> 0 Then

```
Else
    MsgBox "Debe seleccionar el Cliente", vbInformation, "SAVS"
    DoCmd.GoToControl "ID_POL"
    DoCmd.GoToControl "ID_CLIENTE"
End If
End Sub
```

```
Private Sub ID_POL_AfterUpdate()
DoCmd.Requery "FORMA_PAGO"
End Sub
```

```
Private Sub ID_POL_KeyPress(Keycode As Integer)
If Keycode >= 0 And Keycode <= 255 Then
    Keycode = 0
    Beep
End If
```

```
End Sub
```

```
Private Sub ID_PROMOTOR_KeyPress(Keycode As Integer)
If Keycode >= 0 And Keycode <= 255 Then
    Keycode = 0
    Beep
End If
```

```
End Sub
```

BotonCerrar - 1

Option Compare Database

Option Explicit

```
Private Declare Function GetSystemMenu Lib "user32" (ByVal hwnd As Long, ByVal bRevert As Long) As Long
```

```
Private Declare Function GetMenuItemCount Lib "user32" (ByVal hMenu As Long) As Long
```

```
Private Declare Function DrawMenuBar Lib "user32" (ByVal hwnd As Long) As Long
```

```
Private Declare Function RemoveMenu Lib "user32" (ByVal hMenu As Long, ByVal nPosition As Long, ByVal wFlags As Long) As Long
```

```
Const MF_BYPOSITION = &H400&
```

```
Const MF_REMOVE = &H1000&
```

```
Public Sub EliminaBotonCerrar(Frm As Form)
```

```
    Dim WinWnd As Long, Ret As String, RetVal As Long, lpClassName As String
```

```
    WinWnd = Application.hWndAccessApp
```

```
    If WinWnd = 0 Then MsgBox "Couldn't find the window ...": Exit Sub
```

```
    Dim hSysMenu As Long, nCnt As Long
```

```
    hSysMenu = GetSystemMenu(WinWnd, False)
```

```
    If hSysMenu Then
```

```
        nCnt = GetMenuItemCount(hSysMenu)
```

```
        If nCnt Then
```

```
            RemoveMenu hSysMenu, nCnt - 1, MF_BYPOSITION Or MF_REMOVE
```

```
            RemoveMenu hSysMenu, nCnt - 2, MF_BYPOSITION Or MF_REMOVE
```

```
            DrawMenuBar Frm.hwnd
```

```
        End If
```

```
    End If
```

```
End Sub
```

ModuloBas - 1

Option Compare Database

```
Declare Function ShellExecute Lib "shell32.dll" Alias "ShellExecuteA" (ByVal hwnd As Long, ByVal  
    lpOperation As String, ByVal lpFile As String, ByVal lpParameters As String, ByVal lpDirectory  
As String, ByVal nShowCmd As Long) As Long  
Public Const SW_SHOW = 1
```

Option Compare Database

Sub ValSalir()

```
If CurrentProject.AllForms("AGREGAR_BENEFICIARIO").IsLoaded = True Or CurrentProject.AllForms("AGREGAR_COBERTURA").IsLoaded = True Or CurrentProject.AllForms("AGREGAR_CUOTA").IsLoaded = True Or CurrentProject.AllForms("BANCOS").IsLoaded = True Or CurrentProject.AllForms("BENEFICIARIOS").IsLoaded = True Or CurrentProject.AllForms("CANCEL_RECIBO").IsLoaded = True Or CurrentProject.AllForms("CANCEL_RECIBO1").IsLoaded = True Or CurrentProject.AllForms("CLIENTES").IsLoaded = True
```

```
Or CurrentProject.AllForms("COBERTURAS").IsLoaded = True Or CurrentProject.AllForms("COTIZACION_IMPRIMIR").IsLoaded = True Or CurrentProject.AllForms("COTIZACION_NUEVA").IsLoaded = True Or CurrentProject.AllForms("COTIZACIONES").IsLoaded = True Or CurrentProject.AllForms("CUOTAS").IsLoaded = True Or CurrentProject.AllForms("FORMA_PAGO").IsLoaded = True Or CurrentProject.AllForms("INFO_EMPRESA").IsLoaded = True Or CurrentProject.AllForms("INGRESAR_TASA").IsLoaded = True Or CurrentProject.AllForms("MODIFICAR_BANCO").IsLoaded = True Or CurrentProject.AllForms("MODIFICAR_BENEFICIARIO").IsLoaded = True Or CurrentProject.AllForms("MODIFICAR_CLIENTE").IsLoaded = True Or CurrentProject.AllForms("MODIFICAR_COBERTURA").IsLoaded = True Or CurrentProject.AllForms("MODIFICAR_FORMA_PAGO").IsLoaded = True Or CurrentProject.AllForms("MODIFICAR_POLIZA").IsLoaded = True Or CurrentProject.AllForms("MODIFICAR_PROMOTOR").IsLoaded = True Or CurrentProject.AllForms("MODIFICAR_PROVEEDOR").IsLoaded = True Or CurrentProject.AllForms("MODIFICAR_TASA").IsLoaded = True Or CurrentProject.AllForms("MODIFICAR_TIPO_ID").IsLoaded = True Or CurrentProject.AllForms("MODIFICAR_TIPO_PAGO").IsLoaded = True Or CurrentProject.AllForms("MODIFICAR_VENTA").IsLoaded = True Or CurrentProject.AllForms("POLIZA_COMPRA").IsLoaded = True Or CurrentProject.AllForms("PROMOTORES").IsLoaded = True Or CurrentProject.AllForms("PROVEEDORES").IsLoaded = True Or CurrentProject.AllForms("TIPO_ID").IsLoaded = True
```

```
Or CurrentProject.AllForms("TIPO_PAGO").IsLoaded = True Or CurrentProject.AllForms("VENTAS").IsLoaded = True Then
```

```
CommandBars("Barra").Controls(1).Enabled = False
```

```
Else
```

```
CommandBars("Barra").Controls(1).Enabled = True
```

```
End If
```

```
End Sub
```

BIBLIOGRAFÍA

- Folleto de Sistemas Informáticos, Ing. Edgard Robleto.
- Folleto de Sistemas de Apoyo a la Toma de Decisiones, Ing. Edgard Robleto.
- Folleto de Sistemas de Información para Ejecutivos, Ing. Edgard Robleto.
- Folleto de Bases de Datos y DBMS, Ing. Edgard Robleto.
- Sistemas Operativos, Diseño e Implementación, 2da. Edición, Andrew S. Tanenbaum.
- Ingeniería de Software, Un Enfoque Practico, 4ta. Edición, Roger S. Pressman.
- Análisis y Diseño Detallado de Aplicaciones, Mario G. Piattini, Joaquín Cervera, Luis Fernández.
- http://www.osmosislatina.com/proyectos_sistemas.htm
- [http://www.lafacu.com/ análisis estructurado.htm](http://www.lafacu.com/análisis_estructurado.htm)