

**SISTEMA DE INFORMACION PARA LA ADMINISTRACION Y GESTION
PRIMARIA DE LA MICROEMPRESA DIPRECO LTDA, IMPLEMENTANDO LAS
FUNCIONES BASICAS DE INVENTARIOS, CUENTAS POR COBRAR Y NOMINA**

IVETHE ROCIO MONROY BLANCO

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE CIENCIAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA 2007**

**SISTEMA DE INFORMACION PARA LA ADMINISTRACION Y GESTION
PRIMARIA DE LA MICROEMPRESA DIPRECO LTDA, IMPLEMENTANDO LAS
FUNCIONES BASICAS DE INVENTARIOS, CUENTAS POR COBRAR Y NOMINA**

IVETHE ROCIO MONROY BLANCO

**Proyecto de Grado para optar al título de
Ingeniero de Sistemas**

Director

JOSE CARCAMO SEPULVEDA

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE CIENCIAS FÍSICO MECÁNICAS
ESCUELA DE INGENIERÍA DE SISTEMAS E INFORMÁTICA
BUCARAMANGA 2007**

CONTENIDO

	Pág.
1 ESPECIFICACIONES DEL PROYECTO	9
1.1 TITULO.....	9
1.2 MODALIDAD	9
1.3 INDICATIVO DEL TRABAJO.....	9
1.4 AREA DE APLICACIÓN	9
1.5 AUTOR.....	9
1.6 DIRECTOR.....	9
1.7 ENTIDAD INTERESADA.....	9
2 PLANTEAMIENTO DEL PROBLEMA	10
2.1 DESCRIPCIÓN DEL PROBLEMA.....	10
2.2 JUSTIFICACIÓN	11
2.3 OBJETIVOS	11
2.3.1 Objetivo General.....	11
2.3.2 Objetivos Específicos	11
2.4 ALCANCE DEL PROYECTO.....	12
3 MARCO DE REFERENCIA.....	13
3.1 MARCO TEÓRICO.....	13
3.2 MARCO CONCEPTUAL.....	14
3.2.1 Arquitectura del Software	14
3.2.2 Lenguaje de Modelado	16
3.2.3 Tecnología empleada.....	18
4 METODOLOGIA.....	23
4.1 MODELO DE DESARROLLO.....	24
4.2 ANALISIS DE REQUERIMIENTOS.....	24
Requerimientos de usuario.....	24
4.3 DISEÑO GLOBAL	25
4.3.1 Casos de uso	26
4.3.2 Diagramas de Actividades.....	29
4.4 DISEÑO DE PROTOTIPOS	35
4.4.1 Prototipo básico de requerimientos	35
4.4.2 Prototipo de interfaces.....	36
4.4.3 Prototipo de inserción de cifras	36
4.4.4 Prototipo de análisis de información.....	36

5	IMPLEMENTACION.....	38
5.1	MODELO DE DATOS DEL PROYECTO.....	38
5.2	ESPECIFICACIÓN DE LA INTERFAZ.....	39
5.2.1	Acceso al programa.....	39
5.2.2	Inicio de sesión.....	39
5.2.3	Ventana de Menú Principal	41
5.2.4	Submódulo Inventario.....	41
5.2.5	Submódulo Ventas	45
5.2.6	Submódulo Contratos.....	56
5.2.7	Submódulo Actividades	60
5.2.8	Reportes.....	62
5.2.9	Consultas	66
6	PRUEBAS Y REVISION DE PROTOTIPOS	68
6.1	PROTOTIPO BASICO DE REQUERIMIENTOS	69
6.1.1	Descripción de la prueba.....	69
6.1.2	Resultados de la prueba.....	69
6.2	PROTOTIPO DE INTERFACES.....	70
6.2.1	Descripción de la prueba.....	70
6.2.2	Resultados de la prueba.....	70
6.3	PROTOTIPO DE INSERCIÓN DE CIFRAS.....	71
6.3.1	Descripción de la prueba.....	71
6.3.2	Resultado de la prueba	71
6.4	PROTOTIPO PARA ANALISIS DE INFORMACION	71
6.4.1	Descripción de la prueba.....	71
6.4.2	Resultados de la prueba.....	72
6.5	PRUEBA FINAL.....	72
6.5.1	Descripción de la prueba.....	72
6.5.2	Resultado de la prueba	72
7	VERIFICACION DEL CUMPLIMIENTO DE LOS OBJETIVOS.....	73
8	CONCLUSIONES.....	74
9	RECOMENDACIONES	75

BIBLIOGRAFIA

LISTA DE TABLAS

	Pág.
Tabla 1. Elementos de Diagramas UML utilizados en el Diseño.	18
Tabla 2. Clase de Acceso a datos de ADO.NET	22
Tabla 3. Recursos de las pruebas	69
Tabla 4. Cumplimiento de objetivos	76

LISTA DE ANEXOS

	Pág.
Anexo A. Documentación del Modelo de Datos	76

GLOSARIO

DataGrid o Grilla: Cuadrícula desplazable en la cual se muestran datos según consultas realizadas en la base de datos. Esta cuadrícula permite editar, adicionar o eliminar datos.

Crystal Reports: es la herramienta de elaboración de formas estándar para visual estudio .NET. Permite crear contenido interactivo con calidad de presentación.

Usuario: persona que interactúa con el sistema.

Lenguaje Unificado de Modelado (UML): lenguaje estándar para modelado de software, es un lenguaje para visualizar, especificar, construir y documentar los componentes de un sistema con gran cantidad de software.

Dato: cifra o hecho aislado que no ha sido analizado o interpretado.

Dipreco Ltda: Diseño, Prefabricado y Construcción. Fábrica de prefabricados para la construcción.

Salario a destajo: o por unidad de obra, consiste en pagar cierta cantidad por cada pieza o unidad procesada o fabricada por el trabajador. Los trabajadores de la construcción los denominan “al contrato”, y en ella el trabajador no está obligado a cumplir la jornada ordinaria establecida.

RESUMEN EN ESPAÑOL

TITULO: SISTEMA DE INFORMACION PARA LA ADMINISTRACION Y GESTION PRIMARIA DE LA MICROEMPRESA DIPRECO LTDA, IMPLEMENTANDO LAS FUNCIONES BASICAS DE INVENTARIOS, CUENTAS POR COBRAR Y NOMINA

AUTORES: MONROY BLANCO, Ivethe Rocio

PALABRAS CLAVES: Inventario, Productos, Movimientos, Ventas, Producción, Clientes, Cuentas, Contratos.

DESCRIPCIÓN: El sistema ofrece un módulo de administración dividido en 4 secciones: Inventario, Ventas, Contratos y Actividades. En cada uno de estas se implementan las funciones pertinentes para consultar, actualizar o editar registros, ya sea de productos, movimientos en el inventario, pedidos, cuentas por cobrar, producción y contratos de mano de obra, dependiendo de las necesidades. El usuario tiene como prioridad parametrizar, crear los registros correspondientes a las actividades diarias (ya sea en la planta de producción o ventas) y mantener actualizada la base de datos. El usuario puede generar cotizaciones, facturas, contratos a pagar (lo que correspondería a la nómina de pago a destajo) e informes basados en consultas predeterminadas, cuyo propósito principal es la actualización del inventario, las cuentas por cobrar y el seguimiento de los salarios devengados por cada trabajador con el sistema de pago a destajo. Adicionalmente, se crearon opciones de realización de informes y copias de seguridad de la base de datos.

El diseño e implementación del software se basó en los requerimientos descritos por el usuario final, diagramas de casos de uso, diagramas de actividades, observaciones y cambios resultantes de las pruebas según la metodología de prototipado evolutivo y el Lenguaje de modelado unificado UML.

RESUMEN EN INGLÉS

TITLE: INFORMATION SYSTEM FOR THE ADMINISTRATION AND PRIMARY MANAGEMENT OF MICROCOMPANY DIPRECO LTDA., IMPLEMENTING THE BASIC FUNCTIONS OF INVENTORIES, OUTSTANDING ACCOUNTS, AND PAYROLL*

AUTHORS: MONROY BLANCO, Ivethe Rocio **

KEY WORDS: Inventory, Products, Movements, Sales, Production, Clients, Accounts, Information

DESCRIPTION: The system offers a module of administration divided in 4 sections: Inventory, Sales, Contracts and Activities.

In each one of these, the pertinent functions are implemented to consult, to update or to publish registries, whether about products, movements in the inventory, orders, outstanding accounts, production, or contracts of manual labor, depending on the necessities. The user that manage the system, must, as a priority, set up the parameters to create the registries corresponding to the daily activities (in the production plant or sales) and to maintain updates to the data base.

The user can generate quotations, invoices, contracts to pay (which would correspond to the payroll with the system of pay by the job) and information based on predetermined consultations, whose main purpose is the updating of inventory, outstanding accounts and the chase of the salaries earned by each employee. Additionally, options of accomplishment of information and backups were created.

The design and implementation of this software was developed with UML (Unified Modeling Language) diagrams like use case diagrams, activities diagrams and innovations resulting from the tests according to the evolutionary prototyping and requirements of the user.

* Graduation Project

** Physics and Mechanics Sciences Faculty
Systems Engineer School
Director: Jose Carcamo Sepulveda

INTRODUCCIÓN

Este documento recopila información relacionada con la elaboración del trabajo de grado denominado “SISTEMA DE INFORMACION PARA LA ADMINISTRACION Y GESTION PRIMARIA DE LA MICROEMPRESA DIPRECO LTDA, IMPLEMENTANDO LAS FUNCIONES BASICAS DE INVENTARIOS, CUENTAS POR COBRAR Y NOMINA”, el cual surge como respuesta a algunas necesidades identificadas en el manejo de información en dicha microempresa.

El presente proyecto, buscará resolver problemas puntuales para lograr hacer más eficaz y eficiente el proceso de control de inventarios, cuentas por cobrar, y nómina con el sistema de pago a destajo, aprovechando los recursos informáticos para la optimización de procesos fundamentales en la administración de la empresa.

1 ESPECIFICACIONES DEL PROYECTO

1.1 TITULO

Sistema de información para la administración y gestión primaria de la microempresa Dipreco Ltda., implementando las funciones básicas de inventarios, cuentas por cobrar y nomina.

1.2 MODALIDAD

Investigación

1.3 INDICATIVO DEL TRABAJO

Original

1.4 AREA DE APLICACIÓN

Sistemas de información

1.5 AUTOR

Ivethe Rocio Monroy Blanco

1.6 DIRECTOR

Ing. José Cárcamo Sepúlveda

1.7 ENTIDAD INTERESADA

Dipreco Ltda.

2 PLANTEAMIENTO DEL PROBLEMA

Esta fase de investigación preliminar constituye una de las bases fundamentales en el desarrollo de cualquier software, ya que permite entender el origen del problema detectado y plantear una posible solución que contribuya a mejorar el manejo de la información, empleando las herramientas que la tecnología pone al alcance de todos.

2.1 DESCRIPCIÓN DEL PROBLEMA

Dipreco Ltda. produce diariamente prefabricados para la construcción tales como vigas, postes, escaleras, bloque etc. A medida que se van fabricando estos productos se van acumulando en la bodega a menos que sean requeridos inmediatamente por un comprador; la cantidad fabricada depende de factores como demanda, remanente deseado en bodega y el clima (en días soleados los productos pre-fabricados completan rápidamente su fase de secado). En la elaboración del producto se llevan a cabo varias actividades que influyen en el costo de producción del mismo; dichas actividades son costos variables que sumados a otros factores (como desperdicio, precios de materiales etc.) determinan el precio final de venta. En Dipreco, diariamente al final del día o durante el mismo se van registrando estas actividades realizadas en la planta de producción, ya sea fundida de material, transporte, carga etc. Como la empresa utiliza el sistema de pago a destajo a sus empleados, se debe anotar asignando cada una de estas actividades con sus respectivos valores a la persona que la desempeñó. Estas se van sumando a un 'contrato', que consiste en la relación de todas estas actividades, las cantidades y los productos correspondientes. El contrato tiene duración de una semana con opción de alargue, y al final de la misma determinará el valor total de la nómina a pagar, y que dependerá de las labores realizadas durante el intervalo que dure dicho contrato; los empleados que realizan dichas labores reciben semanalmente el pago correspondiente a su desempeño individual durante este tiempo, sumado a un salario básico pactado. El nivel de producción es constante en términos generales, pero puede variar dependiendo de la demanda.

Periódicamente se lleva un control de las existencias haciendo un conteo manual del material que se encuentra en bodega; este dato es anotado en papeles para al final de la semana ser cotejado con la producción y las ventas realizadas. El control de las ventas se lleva en hojas de cálculo donde se anota el nombre del cliente, el valor de la venta, el producto vendido y la posible cuenta por pagar adquirida por el cliente. Los datos de producción y ventas no se encuentran relacionados por lo que deben ser constantemente comparados para saber si existe correspondencia con el movimiento en la planta de producción y la bodega.

Este sistema resulta insuficiente ya que trabaja con los registros de datos relacionados con las ventas y la producción de una forma manual (multiplicidad de papeles y hojas sueltas con información, además de un gran número de archivos de Excel); dificultando el control de la información actualizada del movimiento del inventario y requiriendo la entrada de un mismo dato en diferentes tablas no relacionadas.

2.2 JUSTIFICACIÓN

La implementación de tecnologías de información ayudan a las empresas a mejorar la eficiencia de las líneas de producción, mejorar los procesos productivos, reducir costos, elevar la calidad del bien o servicio que se ofrece, mejorar la atención al cliente y elevar la productividad.

Éste Sistema mejora los procesos de registro de datos que se llevan a cabo a diario en Dipreco Ltda. en el momento de realizar ventas, actualización del inventario y hacer el seguimiento de la mano de obra realizada, puesto que al tener la información en hojas de cálculo y archivos sueltos se pierde tiempo y en ocasiones dinero cuando no quedan registros de deudas anteriores adquiridas por los clientes y cuando no se tiene un dato exacto de las existencias disponibles para la venta y es necesario remitirse a búsquedas infructuosas en una cantidad de tablas distintas de Excel u hojas sueltas con información, a su vez, al hacerse los registros, un mismo dato deba ser digitado en diferentes hojas de calculo, esto puede llevar a errores tales como que en cualquier momento un dato sea omitido y por consiguiente, perderse la información o perjudicar la fiabilidad de la misma.

En Dipreco Ltda., al realizar los procesos de la forma ya descrita, se genera una pérdida considerable de tiempo en el momento de realizar consultas sobre disponibilidad de productos en bodega, información de los clientes, cuentas pendientes, producción, pedidos con entregas pendientes, etc. debido a la dispersión de la información en múltiples fuentes.

2.3 OBJETIVOS

2.3.1 Objetivo General

Implementar un sistema de información para la administración y gestión primaria de la micro empresa Dipreco Ltda., que incluya las funciones básicas de inventarios, cuentas por cobrar y nómina.

2.3.2 Objetivos Específicos

- Modelar una solución informática a los problemas de procesamiento y organización de la información en la microempresa Dipreco Ltda, mediante la integración de conceptos de programación y teoría de Bases de Datos, aplicables en un lenguaje Visual y en una herramienta de gestión de base da datos.

- Desarrollar una herramienta software que facilite el proceso de control de inventarios, el seguimiento de las cuentas por cobrar y el registro de los salarios devengados por cada empleado

2.4 ALCANCE DEL PROYECTO

El software facilita el manejo específicamente de las funciones básicas de inventarios, cuentas por cobrar y la nómina de pago a destajo. Recopila información necesaria para el cálculo y elaboración de informes sobre la producción, ventas a contado y a crédito para efectos de control de salida y entrada de productos y registro de cuentas por cobrar, pedidos de entrega inmediata y pedidos con entregas pendientes, movimientos diarios del inventario, disponibilidad de productos, cotizaciones, facturas, cuentas por cobrar, actividades diarias en la planta de producción y salarios a pagar. El registro diario de las diferentes actividades en el proceso de producción, transporte, etc., realizadas en la fábrica generara al final de la semana, o en la fecha que el usuario lo requiera, un informe con los detalles y el valor del contrato a pagar.

El tipo de usuario que manejará el sistema es el usuario administrador, quien será el encargado de la seguridad del sistema y de la creación, actualización y eliminación de registros en la base de datos.

3 MARCO DE REFERENCIA

El marco de referencia establece las bases conceptuales que definen el software, los fundamentos de los Sistemas de Información, la definición de la arquitectura de la aplicación para llevar un seguimiento constante del software en fases posteriores de desarrollo y, la experiencia de la persona encargada de llevar el registro de las actividades anteriormente mencionadas.

3.1 MARCO TEÓRICO

Dipreco (diseño, prefabricado y construcción) es una microempresa santandereana fundada hace 25 años, dedicada a la prestación de servicios para el área de la construcción encargada de la producción de prefabricados en mortero y concreto reforzado, tales como bloques de concreto liso o abuzardado, gramoquín, adoquines, placas, bancas, calados, postes, vigas etc.

La prefabricación es un método industrial de producción de elementos o partes de una construcción en planta o fábrica y su posterior instalación o montaje en la obra.

Esta técnica, que ha tenido un enorme desarrollo a nivel mundial, presenta claras ventajas cuando se requiere utilizar elementos repetitivos y/o industrializar las faenas de construcción y mejorar su productividad. Entre ellas se destacan las siguientes:

1. Uso múltiple y repetitivo de encofrados o moldajes.
2. Reducción de plazos de construcción.
3. Organización similar a una fábrica, con mayor grado de mecanización, mano de obra estable y especializada, aprovechamiento de las ventajas de la normalización y producción en serie.
4. Mayor facilidad para un adecuado control de calidad.
5. Menor formación de juntas de hormigonado.
6. Posibilidad de aplicar técnicas de pretensado, curado acelerado, etc.

La prefabricación puede llegar a ser aplicada a elementos de hormigón simple, como soleras, tubos, bloques, ladrillos, etc., elementos sencillos de hormigón armado como postes y pastelones, y a sistemas más sofisticados como losas, vigas, columnas. Sus usos varían desde elementos prefabricados para baños, lavaderos, escaleras, hasta techos, andenes, y pavimentos. Algunas de las ventajas del uso de materiales prefabricados son:

- La rapidez de montaje y mantenimiento. Tanto en el caso de viviendas modulares como en edificios con parte de sus componentes prefabricados, se acorta el tiempo de ejecución de la obra, en algunos casos a la mitad, aunque

resulta más espectacular en el caso de viviendas totalmente construidas en fábrica.

- Ahorro de materiales utilizados en obra. Con los elementos prefabricados es posible la construcción o montaje en seco, es decir, se puede prescindir del mortero.
- Reducción de los residuos de la construcción. Al llegar los componentes prefabricados a obra, se reduce la cantidad de residuos generados en el terreno donde se realiza la construcción (embalajes, rotura de piezas...).
- Los componentes prefabricados se construyen con materiales de la misma calidad que en la obra o incluso de calidad superior, puesto que los componentes y proceso de producción están sujetos a un mayor control en la planta (condiciones ambientales, controles de calidad...). Además, los materiales están menos expuestos a inclemencias meteorológicas, ya que el montaje se realiza en un breve período de tiempo y no estarían expuestos a condiciones climáticas adversas durante meses. También poseerían una mayor fortaleza para resistir durante el transporte y montaje.

Actualmente a escala nacional existen aproximadamente 465 empresas prefabricadoras formales, distribuidas en siete zonas, las cuales presentan diferentes tipos de tecnificación y niveles de calidad de sus productos.

Dipreco aspira a convertirse en una microempresa que cubra parte de la demanda de prefabricado a nivel nacional e internacional. Para ello, esta en el proceso de cambio de planta física a una de mayor capacidad para aumentar la producción, también se está estudiando la posibilidad de adquirir nuevas máquinas con las que será posible no solo aumentar la capacidad de producción sino la línea de productos ofrecidos.

3.2 MARCO CONCEPTUAL

3.2.1 Arquitectura del Software

Un objetivo del diseño del software es derivar una representación arquitectónica de un sistema. Esta representación sirve como marco de trabajo para establecer las características y vínculos de cada uno de los elementos que conforman el sistema, en las que se llevan a cabo actividades de diseño más detalladas. Este objetivo es importante en el desarrollo del sistema y la forma de conseguir los requerimientos, la confiabilidad, la seguridad y la habilidad de reutilización de los bloques de construcción arquitectónicos del mismo.

Dada la especificación de las propiedades del diseño arquitectónico es posible su representación mediante modelo. Para el caso particular, la metodología adoptada plantea el establecimiento de estrategias de desarrollo que acompañen la tecnología multicapa de la ingeniería del software (Proceso, métodos y herramientas).

El modelo seleccionado "Prototipado evolutivo" como en los otros, trata de elegir la estrategia adecuada según la naturaleza del proyecto y la aplicación, métodos y

herramientas a utilizarse, y los controles y las entregas requeridas; de acuerdo al bucle de definición de problemas, desarrollo técnico, integración de soluciones y estado actual que caracteriza el desarrollo del software.

▪ **Prototipado Evolutivo**

El prototipo consiste en la elaboración de un prototipo o maqueta del sistema que se construye para evaluar mejor los requisitos que desea que cumpla. Estos modelos suelen consistir en versiones reducidas, demos o conjuntos de pantallas que no son totalmente operativos de la aplicación pedida.

El prototipado evolutivo esta relacionado con un ciclo de vida iterativo en este caso, en vez de seguir el procedimiento habitual que deshecha el prototipo una vez probado y empezar a desarrollar la aplicación, el prototipo supone una primera versión del sistema con funcionalidad limitada. A medida que se comprueba si las funciones implementadas son las apropiadas, se corrigen, refinan o se añaden otras nuevas hasta llegar al sistema final.

La cualidad esencial de un prototipo debe ser la posibilidad de ser construido más rápidamente que la aplicación correspondiente.

Las herramientas utilizadas en el prototipado pueden ser muy variadas, en el primer nivel se encuentran herramientas comunes y fácilmente disponibles como programas de dibujo o de presentación, hojas de cálculo o generadores de informes. El usuario puede ver como quedará la entrada/salida de la aplicación cuando este acabada.

En un nivel más evolucionado se situarían algunos gestores de bases de datos y sistemas de cuarta generación que permiten prototipos más sofisticados que incluyen no sólo interfaces, sino también prototipado del manejo de datos.

Otra opción son las posibilidades de prototipado que proporcionan ciertas herramientas CASE o determinado generadores de aplicaciones. En este caso, se pueden reutilizar las plantillas de pantallas, las asignaciones de teclas de funciones, etc. Para continuar el desarrollo con la misma herramienta.

Se calcula que el prototipado puede consumir el 10 % del presupuesto del proyecto y puede llevar unos pocos días en su desarrollo. En algunos casos se utiliza como un medio para formalizar la aceptación previa del cliente de los requisitos del proyecto.

El uso principal de esta metodología es la ayuda a los clientes y desarrolladores para entender los requerimientos del sistema. El prototipo puede ser usado para entrenamiento antes de que el sistema final sea entregado. El prototipo puede ser utilizado para pruebas. Entre las principales funciones encontramos:

- Se exponen los malentendidos entre los usuarios del software y los desarrolladores.
- Se detectan los servicios que hacen falta.
- Se identifican servicios confusos.
- Un sistema desarrollado esta disponible antes del sistema final.
- El prototipo puede servir como la base de la especificación del sistema.

Figura .Proceso de Prototipado ¹

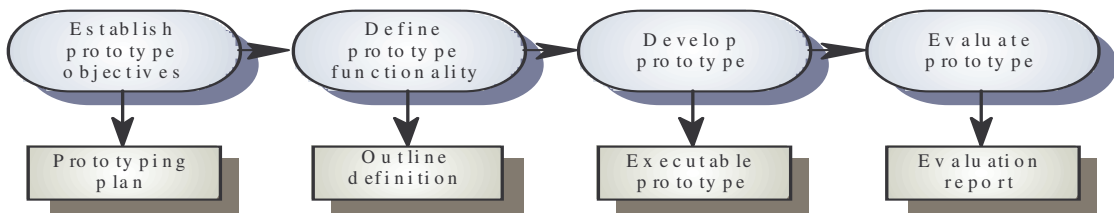
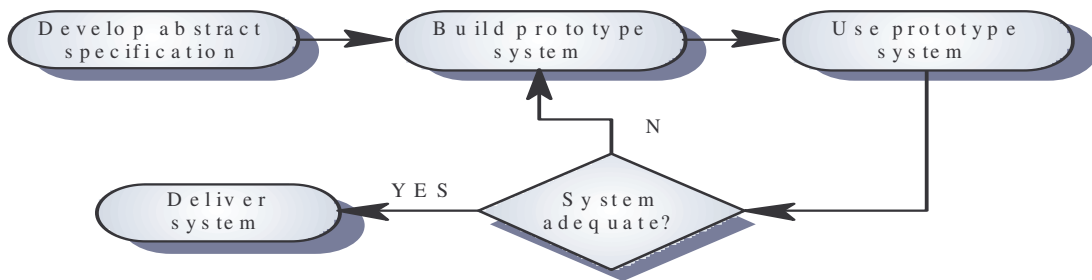


Figura .Diagrama de flujo del prototipado Evolutivo



3.2.2 Lenguaje de Modelado

Existen técnicas y metodologías que pretenden facilitar la construcción de software, el lenguaje de modelado UML hace posible expresar, de una forma bastante gráfica, la interacción de los distintos componentes del Sistema de Información desarrollado y tener un apoyo mas al desarrollo del proyecto.

Grady Booch, Jim Rumbaugh e Ivar Jacobson combinaron y recopilaron las mejores características de cada uno de los métodos de diseño y análisis orientado a objetos en un método unificado, denominado Lenguaje de Modelado Unificado (UML). UML es un lenguaje de especificación, visualización, construcción y documentación de propósito general, especializado en sistemas software. UML permite a un ingeniero de software expresar un modelo de análisis utilizando una notación de modelado con unas reglas sintácticas, semánticas y practicas. La sintaxis nos dice como mostrar y combinar los símbolos. “La sintaxis es

¹ Ian Sommerville. Ingeniería del software Quinta Edición 1995.

comparable a las palabras en el lenguaje natural: es importante saber como se escriben y como combinarlas correctamente para formar una frase. Las reglas semánticas nos dicen lo que significa cada símbolo y como interpretarlo, tanto cuando aparece solo como cuando lo hace en combinación con otros. Es comparable al significado de las palabras en el lenguaje natural. Las reglas prácticas definen el significado de los símbolos a través de los cuales se obtiene el modelo y se hace comprensible para otras personas. Eso correspondería, el lenguaje natural, a las reglas de construcción de frases claras y fácilmente comprensibles” (definición de Ericsson y Pender (ERI98))

En UML un sistema viene representado por cinco vistas diferentes que los describe desde diferentes perspectivas. Cada vista se representa mediante un conjunto de diagramas. En UML están representadas las siguientes vistas:

- Vista del Usuario: representa el sistema (producto) desde la perspectiva de los usuarios (actores). El caso de uso es el enfoque elegido para modelar esta vista.
- Vista Estructural: los datos y la funcionalidad se muestran desde dentro del sistema, es decir, modela la estructura estática (objetos, clases y relaciones)
- Vista del comportamiento: esta parte del modelo del análisis representa los aspectos dinámicos o de comportamiento del sistema. También muestra las interacciones o colaboraciones entre los diversos elementos estructurales en las vistas anteriores.
- Vista de implementación: los aspectos estructurales y de comportamiento se representan aquí tal y como van a ser implementados.
- Vista del entorno: aspectos estructurales y de comportamiento en el que el sistema a implementar se representa.

En general, el modelo de análisis de UML se centra en las vistas del usuario y estructural, siendo estas las utilizadas en el desarrollo de este proyecto.

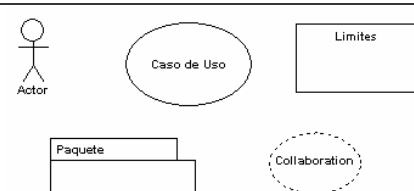
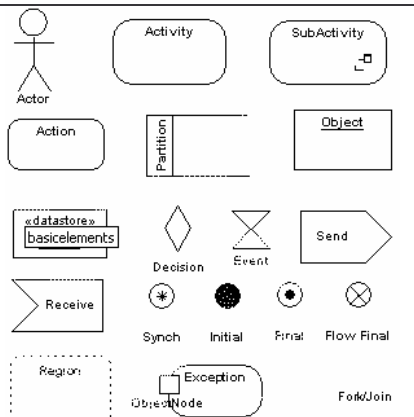
UML, cobra importancia en éste desarrollo porque:

- Es un lenguaje de modelado visual que se usa para especificar, visualizar, construir y documentar los componentes de un sistema. Se emplea para entender, diseñar, configurar, mantener y controlar la información sobre los sistemas a construir.
- No es un lenguaje de programación y es independiente de cualquiera de ellos.
- Es un lenguaje de modelado de propósito general que pueden usar todos los modeladores. No tiene propietario y está basado en el común acuerdo de gran parte de la comunidad informática.
- Puede ser aplicado a diferentes tipos de sistemas (software y no-software), dominios (negocios Vs. Software) y métodos o procesos.

- Se ha convertido en el estándar de facto para definir, organizar y visualizar los elementos que configuran la arquitectura de una aplicación orientada a objetos.

Los elementos correspondientes a los diagramas UML que se han utilizado son:

Tabla 1. Elementos de Diagramas UML utilizados en el Diseño.

DIAGRAMAS UML	NOTACIÓN
Diagramas de Casos de uso	
Diagramas de Actividades	

3.2.3 Tecnología empleada

▪ Visual Basic.Net

La plataforma .NET creada por Microsoft esta diseñada para explotar Internet al máximo y con facilidad, pero sin olvidar las aplicaciones tradicionales. Visual Basic. Net es solo uno de los lenguajes que pueden correr sobre esta plataforma. Desde el punto de vista del desarrollador se puede decir que .Net se compone del framework, Visual Studio .NET y los Net Enterprise Servers.

El Framework.NET

El framework conforma el núcleo de la nueva plataforma. Ésta es la base sobre la cual se desarrollan y se ejecutan programas escritos en Visual Basic.NET, así como en otros lenguajes; para lograr esto es necesario que el marco .NET imponga ciertos aspectos comunes entre estos lenguajes, por lo cual como

desarrolladores de visual Basic, nos vemos beneficiados al adquirir casi por inercia muchas características avanzadas de otros lenguajes (como threading). Es decir si el maraco .NET soporta alguna característica, esta se transfiere a todos los lenguajes que lo utilizan. Este framework o marco está integrado por un gran conjunto de librerías, compiladores y utilidades para el programador. Actualmente las aplicaciones desarrolladas sólo pueden correr bajo Windows, pero hay una versión de “Net Compact Framework”, que permite correr aplicaciones de .NET en dispositivos móviles, incluso existen proyectos Open Source para portar el framework a Linux. Con migrar el framework sin alterar las aplicaciones ya terminadas y compiladas es posible.

Visual Studio.NET

Es el nuevo entorno de desarrollo (IDE) que permite desarrollar visualmente en .NET, cualquiera que sea el lenguaje elegido. En este entorno se pueden desarrollar aplicaciones creando ventanas y escribiendo el código necesario en un ambiente visual cómodo. Es imposible desarrollar con Visual Studio.NET si el framework no está instalado debajo de éste.

NET Enterprise Servers

Forman la familia de servidores Microsoft que son la base de grandes aplicaciones distribuidas, bases de datos, manejo de dispositivos móviles, integración con sistemas legacy, y mucho más.

A continuación se describirán algunas de las características del lenguaje Visual Basic .NET.

Extensión de los ficheros de código.

En Visual Basic .NET a diferencia de lo que ocurría en las versiones anteriores de Visual Basic, sólo existe un tipo de fichero de código, el cual tiene la extensión .vb, en este tipo de fichero pueden coexistir distintos tipos de elementos, por ejemplo: un módulo de clase, un formulario, un módulo de código, un control, etc.; mientras que en las versiones anteriores de Visual Basic, cada uno de estos elementos tenían su propio tipo de fichero con su respectiva extensión.

Tipos de ejecutables.

Con Visual Basic .NET se pueden crear básicamente estos dos tipos de ejecutables: de consola, no gráfico, al estilo del viejo MS-DOS, y gráficos, como los que normalmente estamos acostumbrados a ver en Windows.

Existen otros tipos de aplicaciones que se pueden crear con Visual Basic .NET: aplicaciones ASP.NET, (realmente no es una aplicación o ejecutable, sino un compendio de distintos tipos de elementos...), servicios Web, servicios Windows, etc.

Namespace (o espacio de nombres)

"Un espacio de nombres es un esquema lógico de nombres para tipos en el que un nombre de tipo simple, como MiTipo, aparece precedido por un nombre jerárquico separado por puntos. [...]"

Así es como lo definen en eBook Microsoft .NET Framework, cuya versión en Castellano se puede conseguir usando el link: [http://www.microsoft.com/spain/download/msdn/ebooks/Microsoft.NETFramework\(PPC\).lit](http://www.microsoft.com/spain/download/msdn/ebooks/Microsoft.NETFramework(PPC).lit) .

Un Namespace, es una forma de agrupar clases, funciones, tipos de datos, etc. que están relacionadas entre sí. Por ejemplo, entre los Namespaces que podemos encontrar en el .NET Framework encontramos uno con funciones relacionadas con Visual Basic: Microsoft.VisualBasic. Un Namespace puede estar repartido en varios assemblies o librerías. Por otro lado, un assembly, (o ensamblado), puede contener varios Namespaces. Pero de esto no debe preocuparnos, ya que el IDE de Visual Studio .NET se encarga de "saber" en que assembly está el Namespace que necesitamos.

Assembly (o ensamblado)

"Un ensamblado es el bloque constructivo primario de una aplicación de .NET Framework. Se trata de una recopilación de funcionalidad que se construye, versiona e instala como una única unidad de implementación (como uno o más archivos). [...]"

Podríamos decir que un assembly es una librería dinámica (DLL) en la cual pueden existir distintos espacios de nombres. Un ensamblado o assembly puede estar formado por varios ficheros DLLs y EXEs, pero lo más importante es que todos los ensamblados contienen un manifiesto (o manifest), en lo referente a las distintas versiones de las librerías y ejecutables. Cada assembly contiene un manifiesto en el cual se indica: el nombre y la versión del assembly, si este assembly depende de otros ensamblados, con lo cual se indica hasta la versión de dichos ensamblados, los tipos expuestos por el assembly (clases, etc.), y permisos de seguridad para los distintos tipos contenidos en el assembly. También se incluyen en los assemblies los datos del copyright, etc.

La ventaja de los ensamblados es que realmente no necesitan de una instalación y un registro correcto en el registro del sistema de Windows, ya que es el "intérprete" de .NET el que se encarga de hacer las comprobaciones cuando tiene que hacerlas. Por tanto podríamos distribuir una aplicación sin necesidad de crear un programa de instalación. Pero, si la aplicación usa ensamblados compartidos, puede que sea necesario usar una instalación.

Los ensamblados compartidos se pueden usar por varias aplicaciones diferentes y deben estar "debidamente" instalados en el directorio asignado por el propio .NET Framework. Ejemplo de ensamblados compartidos son los que definen las clases (tipos) usados por el propio .NET Framework.

▪ **Base de Datos**

Para complementar el desarrollo del Sistema Software, es importante hablar de la capa de datos que conforma el sistema y que contiene los datos de los diferentes contribuyentes de los impuestos y usuarios que acceden al sistema, los parámetros legales del cobro tributario de dichos impuestos y el detalle de las transacciones a nivel de cobro de estos impuestos. Estos valores se entrelazan de diversas maneras para permitir la elaboración de informes de cartera, estados financieros, y datos generales de contribuyentes que cada usuario, según sus necesidades requiera del sistema.

La filosofía de las bases de datos relacionales va más allá de la definición de una base de datos como un conjunto de tablas formadas por registros y campos, pues al establecer relaciones los datos cobran sentido real. Microsoft Access 2000, es la base de datos relacional elegida para desarrollar este proyecto. La capa de negocios consulta, modifica y almacena diferentes tipos de valores proporcionados o requeridos por el usuario en la base de datos a través de consultas estructuradas en SQL².

▪ **Acceso a datos desde VB.Net**

En el software a desarrollar es de vital importancia el acceder a una base de datos para hacer consultas, obtener resultados, agregar, modificar o eliminar registros. Para esto contamos con la tecnología de acceso de datos desde cualquier lenguaje de .NET llamada ADO.NET. El objetivo de esta tecnología es proveer un modelo de clases para acceder a una base de datos. Desde ADO es posible especificar un proveedor OLE DB, para conectarse con un origen de datos cualquier, en este proyecto una base de datos Access.

ADO.NET esta diseñado para trabajar desconectado de la base de datos, y esta integrado con XML. En .NET, el namespace con las clases de ADO.NET es *system.Data*. Allí hay clases para acceder a bases de datos, sea por medio de un proveedor OLE DB o a través de un proveedor nativo de .NET para una base específica. La clase *OleDbConnection* representa una conexión a una base de datos (en nuestro caso Access) , por medio de un proveedor OLE DB. Tenemos que

considerar que OLE DB está basado en COM y, por lo tanto, su uso requiere los servicios de Interop(interoperabilidad), lo que produce una penalidad de performance.

En la siguiente tabla vemos las clases principales para acceder a datos a través de un proveedor OLE DB y que son utilizadas en el desarrollo del proyecto.

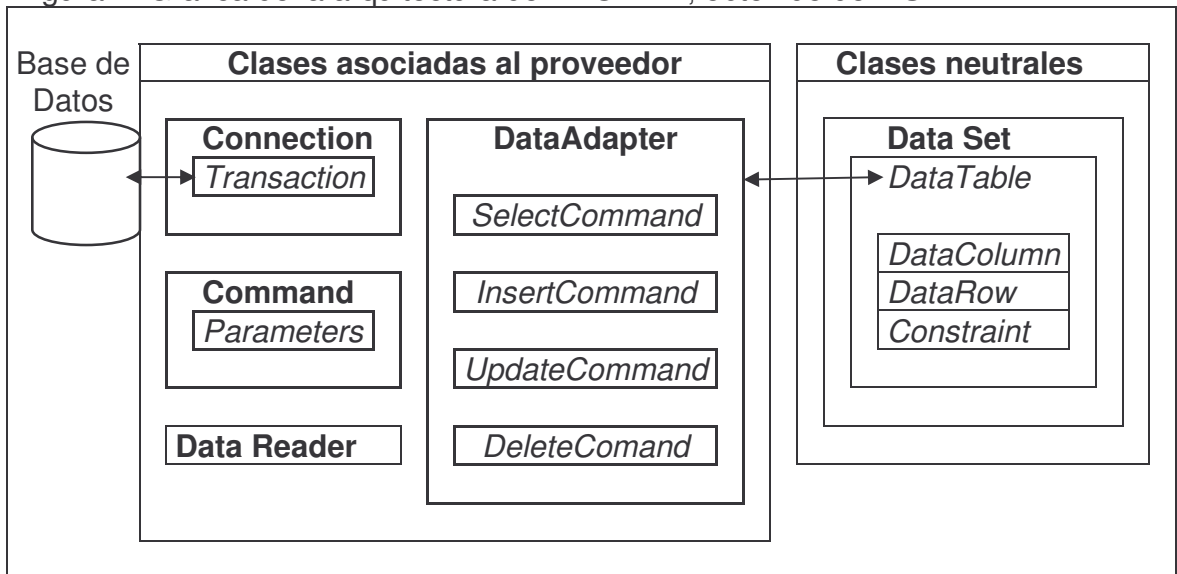
² SQL es un lenguaje universal dedicado a las bases de datos. Es un lenguaje de sintaxis simple y muy potente. Mediante él se puede recorrer, modificar o borrar registros de las tablas de datos.

Tabla 2. Clase de Acceso a datos de ADO.NET

Clase	Descripción
OleDbConnection	Representa la conexión con la base de datos
OleDbCommand	Representa un comando que se puede ejecutar sobre la base de datos.
OleDbDataReader	Es una clase que permite leer los resultados de un comando, de forma secuencial y siempre hacia delante.
OleDbDataAdapter	Agrupar una conexión y varios comandos que se ejecutaran sobre ella, sobre datos que están relacionados
DataSet	Contiene una representación en memoria de datos obtenidos de una base. Por ejemplo, aquí se cargan los resultados de un comando.

El siguiente cuadro muestra la arquitectura de ADO.NET

Figura 1. Grafica de la arquitectura de ADO.NET, obtenido de MSDN.



4 METODOLOGIA

La metodología es considerada como el conjunto de procedimientos, técnicas, herramientas y soporte documental que ayuda a los desarrolladores a realizar nuevo software³. Una metodología puede seguir uno o varios modelos de ciclo de vida, es decir, el ciclo de vida indica qué es lo que hay que obtener a lo largo del desarrollo del proyecto pero no cómo hacerlo; la metodología indica cómo hay que obtener los distintos productos parciales y finales. Algunos de los diferentes modelos de ciclos de vida son:

- *Lineal*: consiste en descomponer la actividad global del proyecto en etapas separadas que son realizadas de manera lineal, es decir, cada etapa se realiza una sola vez, a continuación de la etapa anterior y antes de la etapa siguiente.
- *En cascada*: es un ciclo de vida que admite iteraciones. Después de cada etapa se realiza una o varias revisiones para comprobar si se puede pasar a la siguiente. Es un método rígido, poco flexible y con muchas restricciones.
- *En cascada con subproyectos*: sigue el modelo de ciclo de vida en cascada. Cada una de las cascadas se divide en subetapas independientes que se pueden desarrollar en paralelo.
- *Iterativo*: derivado del ciclo de vida en cascada, este modelo busca disminuir el riesgo que surge entre las necesidades del usuario y el producto final por malos entendidos durante la etapa de solicitud de requerimientos.
- *Incremental*: se realiza construyendo por módulos que cumplen las diferentes funciones del sistema.
- *Espiral*: el modelo se basa en una serie de ciclos repetitivos para ir ganando madurez en el producto final. En cada ciclo (espirales) las especificaciones del producto se van resolviendo paulatinamente.
- *Prototipado evolutivo*: consiste en la construcción de una implementación parcial que cubre los requisitos conocidos, para ir aprendiendo el resto, y , paulatinamente, incorporarlos al sistema. Reduce el riesgo y aumenta la probabilidad de éxito.

³Definiciones tomadas del libro Análisis y Diseño Detallado de Aplicaciones Informáticas. PIATTINI MARIO G, Capítulo 4. Metodologías de Desarrollo de Software.

4.1 MODELO DE DESARROLLO

El prototipado se recomienda en cualquier aplicación que cree pantallas visuales dinámicas, que interactúe intensamente con el ser humano compuesta de algoritmos o procesos que se creen de manera progresiva es recomendado para la creación de un prototipo.

Teniendo esto en mente y en pro del trabajo realizado durante cada etapa del modelo a seguir el modelo de Prototipado Evolutivo se ajusta a las características buscadas del sistema software.

Las metodologías descomponen el proceso hasta el nivel de tareas o actividades elementales. Para cada tarea se identifica un procedimiento que define la forma de ejecutarla y es el vehículo de comunicación entre desarrolladores y usuarios. El modelo de desarrollo permite realizar la especificación de requerimientos, desarrollo de los módulos, implementación, revisión y pruebas, entre otras actividades necesarias para obtener el producto deseado.

Dentro del modelo a llevar, el análisis de requisitos constituye la fase de comunicación mas intensa del proceso de ingeniería del software, esta fase es fundamental para obtener el éxito del producto final que requiere un perfecto entendimiento y comprensión del mismo, en este proceso de comprensión de necesidades y aplicación de principios operativos se obtienen uno o más productos, estos pueden ser productos intermedios, si sirven de base para realizar nuevos productos durante el desarrollo de productos finales, o puede desecharse después de la aprobación del cliente o emplearlo como primera parte de una actividad de análisis a la que seguirá el diseño y la construcción es decir el prototipo se convierte en la primera versión del producto terminado.

4.2 ANALISIS DE REQUERIMIENTOS

Requerimientos de usuario

El usuario administrador, quien será el encargado de poblar la base de datos presenta los siguientes requerimientos:

- Cambiar la clave de acceso al sistema software según lo crea el usuario.
- Contar con una interfaz sencilla, fácil de entender y de manejar.
- Consulta de los detalles de un producto, edición e inserción de nuevos registros.
- Consulta de datos de clientes, edición e inserción de nuevos registros.
- Consulta de datos de empleados, edición e inserción de nuevos registros.
- Elaboración y registro de pedidos donde se calcule el total a pagar según los productos seleccionados junto con los descuentos e impuestos indicados, y la generación de la cuenta por cobrar correspondiente si se

trata de un pedido a crédito. Actualización de los movimientos del inventario descargando productos vendidos.

- Facturación del pedido.
- Generación de cotización.
- Consulta de movimientos en el inventario según fechas y según productos.
- Consulta, registro de pagos y actualización de cuentas por cobrar
- Registro de entregas de pedidos pendientes con posibilidad de entregas parciales. Actualización del inventario
- Registro de salida de productos por concepto de reposición (desperdicio)
- Registro y consulta de contratos
- Registro y consulta de actividades realizadas.
- Registro de productos elaborados. Actualización del inventario
- Consulta de detalles de nómina a cancelar por contrato.
- Consulta de detalles de nómina a cancelar por empleado.
- Consulta de costos de actividades, edición de registros.
- Reporte de los movimientos de inventario.
- Reporte de pedidos y de ventas por producto.
- Reporte de contrato y nómina.
- Reporte de cuentas por cobrar.

4.3 DISEÑO GLOBAL

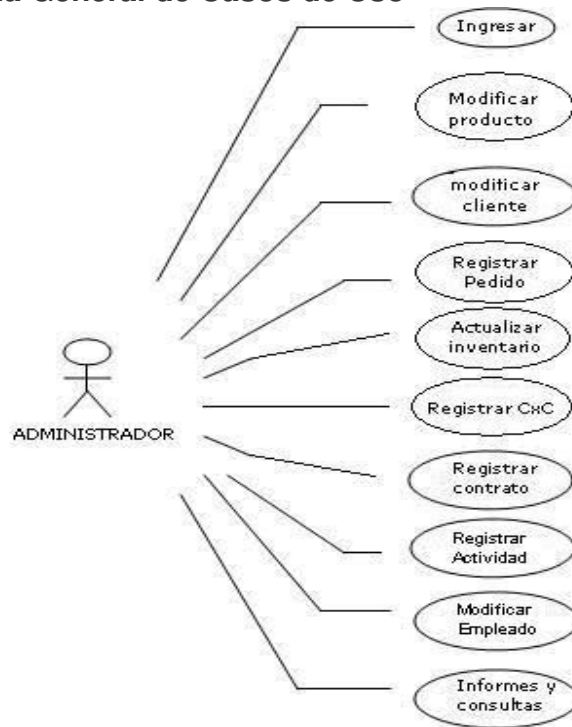
El diseño define estructuralmente el proyecto para soporte y apoyo en la fase de codificación del software. Para diseñar el Sistema se empleó el Lenguaje Unificado de Modelado UML, enfocado en los diagramas de casos de uso y diagramas de actividades.

Partiendo de los objetivos trazados y de los resultados obtenidos en la etapa de análisis de requerimientos se plantean los siguientes submódulos:

- Submódulo Inventarios
- Submódulo Ventas
- Submódulo Contratos
- Submódulo Actividades

En la Figura 2 se muestra el escenario que describe la forma como va a ser usado el software en una determinada situación por medio de diagrama general de casos de uso.

Figura 2. Diagrama General de Casos de Uso



4.3.1 Casos de uso

En la elaboración de casos de uso se tiene en cuenta la interacción de los actores con el caso de uso y la forma excepcional en que el actor puede interactuar con dicho elemento.

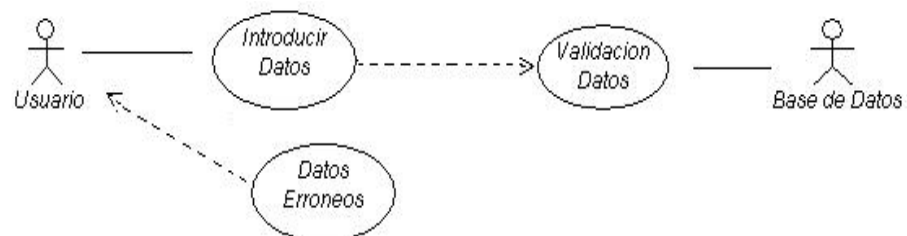
▪ Caso de uso Ingresar

El acceso a sistema se realiza con las siguientes operaciones:

Llenar datos: el usuario llena los datos requeridos para el ingreso.

Datos no validos: es un procedimiento de control para el acceso al sistema.

Figura 3. Caso de uso Ingresar



▪ **Caso de uso Nuevo cliente**

Cuando el usuario intenta registrar un nuevo pedido mediante un formulario se le pregunta si el cliente ya existe, si no es así tiene la opción de agregar uno nuevo.

Nuevo Cliente: se activa la opción de agregar cliente

Introducir datos: se llenan los campos con los datos correspondientes

Datos erróneos: se valida si los datos introducidos fueron correctos

Guardar datos: se actualiza la base de datos.

Figura 6.Caso de uso Nuevo cliente



▪ **Caso de uso Registrar pedido**

Para registrar un pedido nuevo se realizan las siguientes operaciones:

Buscar cliente: si ya existe se selecciona, si no, se guarda uno nuevo.

Escoger productos: se añaden los productos a pedir con sus respectivos precios y cantidades.

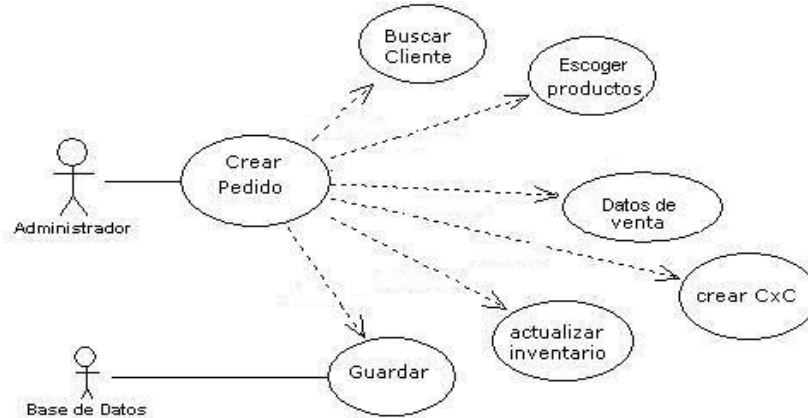
Datos de venta: datos de fechas de pedido y entrega del mismo, venta a crédito o a contado, anticipos etc.

Crear cuenta por cobrar: en el caso de que la venta sea a crédito

Actualizar inventario: descargar del inventario los productos pedidos en caso de que sea un pedido con entrega inmediata.

Guardar Cambios: se crea un pedido nuevo en la base de datos

Figura 6.Caso de uso Registrar pedido



▪ **Caso de uso Actualizar inventario**

El inventario se actualiza automáticamente cada vez que se hacen entregas de pedidos, ya sea el mismo día que se hizo la venta o si se trata de un pedido pendiente. Cuando se registran entradas de productos por producción también se actualiza. El usuario puede actualizar el inventario también en el momento de registrar reposiciones de productos averiados en las entregas.

Figura 7.Caso de uso Actualizar inventario



▪ **Caso de uso Registrar C x C**

Cuando se ha registrado un pedido y el pago es a crédito se genera una nueva cuenta por cobrar

Buscar cliente: se busca el cliente que esta haciendo el pedido

Registrar C x C: se registra el valor de la cuenta

Guardar datos: se guarda el registro correspondiente

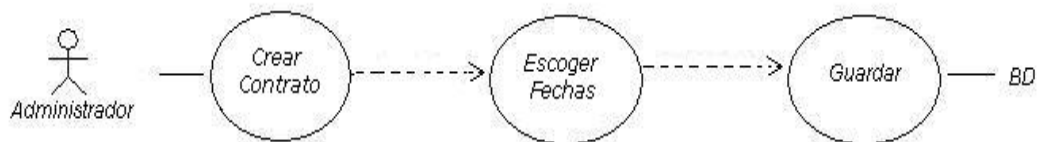
Figura 8. Caso de uso Registrar C x C



▪ **Caso de uso Crear contrato**

Para crear un nuevo contrato se deben escoger las fechas de inicio y terminación del mismo. El contrato nuevo será creado y estará listo para que le sean agregadas actividades.

Figura 9. Caso de uso Registrar C x C



▪ **Caso de uso Realizar informes y consultas**

En este caso de uso el usuario podrá realizar la operación realizar informes o consultas según las necesidades e informes disponibles en el sistema.

Figura 10. Caso de uso Realizar informes y consultas



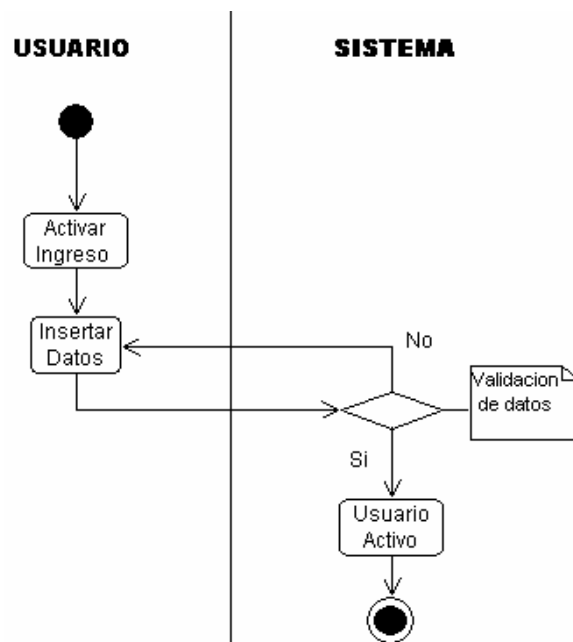
4.3.2 Diagramas de Actividades

Los Diagramas de actividades son usados para modelar el comportamiento de un sistema, y el camino en cuanto este comportamiento este relacionado con el flujo total del sistema. La trayectoria lógica sigue un proceso basado en condiciones, ocurrencia de procesos, acceso a datos, interrupciones y otros; todos estos son usados para construir un proceso, sistema o procedimiento. El diagrama de actividades modela el funcionamiento del sistema y el flujo de control entre los objetos. Son muy similares a los diagramas de flujo, presenta puntos de decisión y operación.

▪ **Diagrama de actividades Ingresar**

En el momento en que el usuario carga el programa, se activa el formulario del ingreso en el cual debe digitar el nombre de usuario y la contraseña que ha elegido en el momento de registrarse. Al hacerlo, el sistema chequea que los valores suministrados sean los correctos, que dicho usuario exista y, si es así, inicia la sesión correspondiente a dicho usuario; de lo contrario, no ingresa al sistema. Este diagrama puede apreciarse en la figura 11.

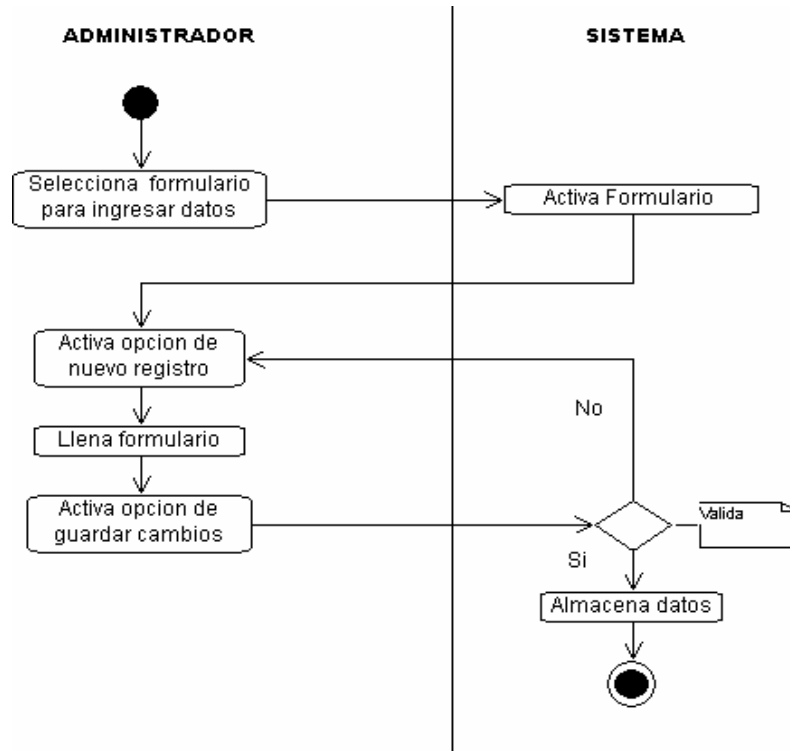
Figura 11. Diagrama de actividad para el caso de uso ingresar



▪ **Diagrama de actividades Nuevo Registro**

En la figura 12 se detallan las actividades que realizan el usuario y el sistema para crear un nuevo registro en la base de datos. Para crear un nuevo registro concerniente al mantenimiento general del sistema siempre se realiza una búsqueda que me relacione el dato que quiero crear con las opciones existentes en el menú mantenimiento, se ingresan los datos en el formulario obtenido por la búsqueda y el sistema valida estos datos para que el administrador revise y de la orden de guardar al sistema para que esté los almacene en la base de datos.

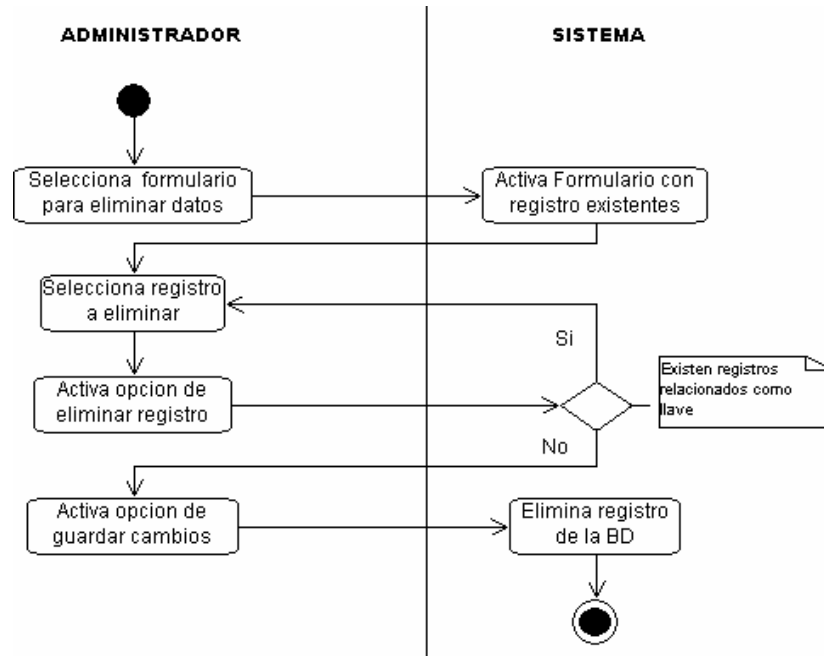
Figura 12. Diagrama de actividades Nuevo Registro



▪ **Diagrama de actividades Eliminar Registro**

En el proceso de eliminación de registros de la base de datos referentes al mantenimiento, el usuario elige el formulario en el que se encuentra dicho registro, el sistema carga los datos correspondientes para que el administrador localice el registro que desea eliminar, después de tener los datos el usuario confirma la eliminación del registro, para que el sistema verifique la no existencia de registros en el que el este aparezca como llave para que el administrador confirme la eliminación y el sistema proceda a borrar los registros relacionados con el mismo. Las actividades necesarias para realizar lo descrito anteriormente se ven en la figura 13

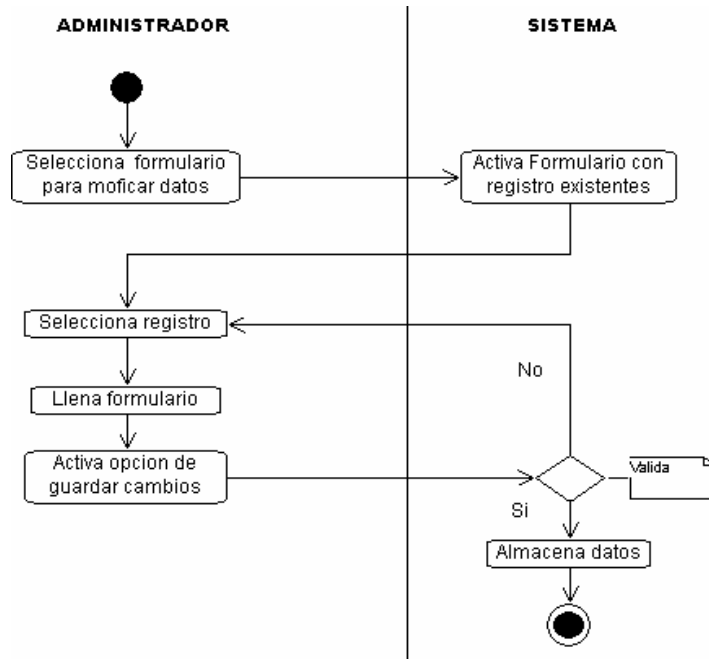
Figura 13. Diagrama de actividades Eliminar Registro



▪ **Diagrama de actividades Modificar Registro**

Aquí, el usuario elige el formulario en el que se encuentra el registro que desea modificar, el sistema carga los datos correspondientes para que el administrador localice los datos a modificar y llene el formulario con los nuevos datos para después el administrador active la opción de guardar cambios y el sistema almacene los cambios en los respectivos registros de la base de datos. Ver figura 14

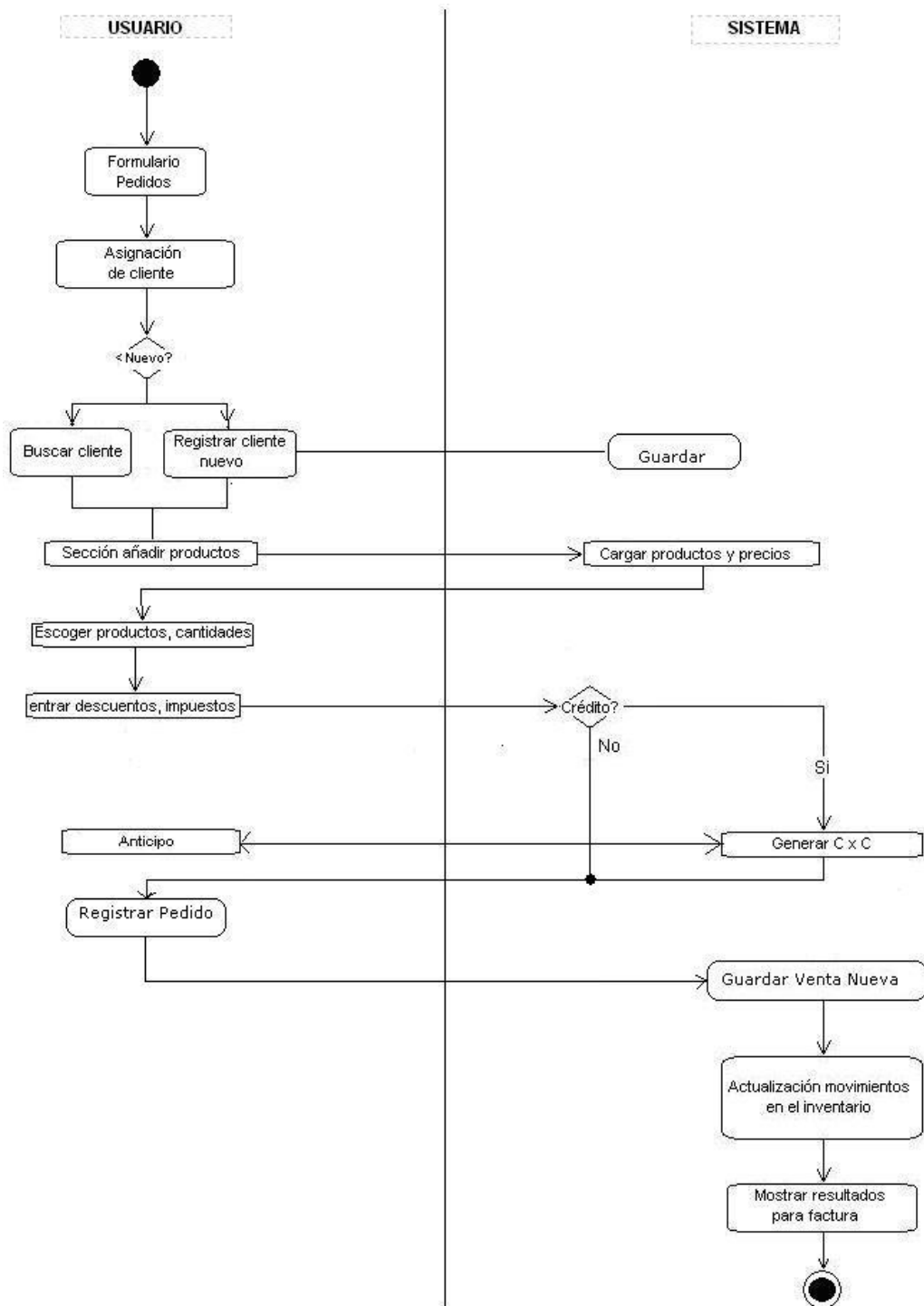
Figura 14. Diagrama de actividades Modificar Registro.



▪ **Diagrama de actividades Registrar Pedido**

El usuario ingresa al formulario para registrar pedido nuevo. Lo primero que debe hacer es asignar un cliente, si es un cliente antiguo lo busca y lo selecciona de una lista, si no, debe registrar el cliente nuevo y llenar los campos correspondientes a los datos del mismo. Luego de esto escoge los productos, que se han cargado con sus respectivos precios. En seguida llena los datos de venta como fecha de pedido y entrega, descuentos, anticipos etc. indica si la venta es a contado a crédito; ya el pedido está listo para ser registrado. Una vez el cliente registre el pedido se genera una venta nueva y todos los detalles del mismo son almacenados. Si la venta fue a crédito, automáticamente se genera una cuenta por cobrar a cargo del cliente seleccionado en el primer paso, después de esto, la cantidad de productos en bodega es modificada, y el registro queda listo para generar una factura.

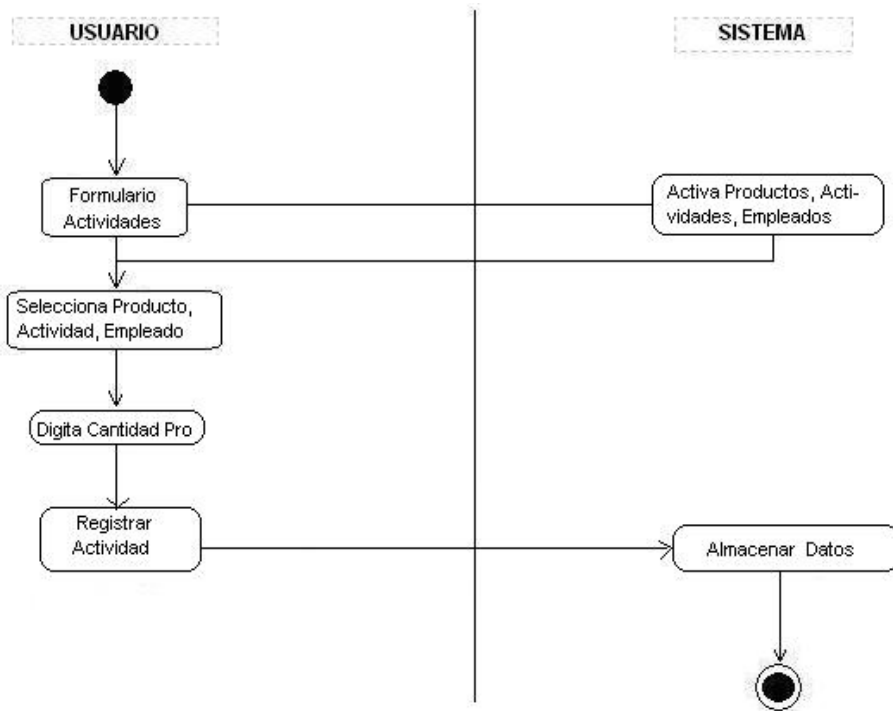
Figura 15. Diagrama de actividades Registrar Pedido



▪ Diagrama de actividades Registrar Actividad

El usuario escoge de una lista el producto, la actividad y el empleado al que será asignada dicha actividad, digita la cantidad de producto correspondiente y registra. La actividad queda registrada en el contrato activo en ese momento y los datos son actualizados.

Figura 16. Diagrama de actividades Registrar Actividad



4.4 DISEÑO DE PROTOTIPOS

Por medio de un estudio detallado de los distintos casos de uso y diagramas de actividades para el sistema y siguiendo el modelo de prototipado evolutivo, se plantean los diferentes prototipos a seguir para el desarrollo del sistema software.

4.4.1 Prototipo básico de requerimientos

Este prototipo fue la primera versión de lo que sería el prototipo de análisis de información.

En este prototipo se planteo una visión global de usuario cuyo propósito principal fuese el registro de los movimientos del inventario, para esto se diseñaron dos formularios: el Formulario de Registro de Pedidos, que crearía un registro nuevo a cargo de un cliente, haría las operaciones de multiplicar valores unitarios por cantidades, calculo de descuentos, IVA, valores parciales y totales y finalmente, descargaría del inventario los productos a vender. El otro formulario sería el de registro de productos elaborados en las actividades diarias con introducción de los mismos al stock del inventario.

En estos formularios el usuario introducía datos necesarios para el calculo del valor total de la venta, tales como: precios unitarios, cantidades, porcentaje de descuentos o impuestos y también introducía la fecha. De igual manera para el formulario de registro de productos elaborados se introducían los productos con sus cantidades correspondientes y el empleado que ejecutó la mano de obra.

Este prototipo tiene como fin aclarar dudas con el usuario sobre lo que podría ser la base de evolución para el producto final sin perder de vista todos sus objetivos, definir los requerimientos hardware y software básicos para la implementación.

4.4.2 Prototipo de interfaces

El desarrollo de este prototipo se realizo sobre el prototipo anterior y basándose en los requerimiento previamente obtenidos y organizados en los diagramas de actividades y casos de uso. El propósito es mostrar las ventanas, su navegación, interacción, controles y botones al usuario y obtener una realimentación que nos permite mejorar el diseño de interfaz respecto a la cantidad de información sobre la pantalla, colores, organización y otros aspectos, usando alta gama de controles y herramientas de .NET. Con los diagramas de actividades se realizó la interfase entre el usuario y la pantalla de visualización, en esta se presenta la información como va ha ser proporcionada al sistema o como es recuperada de este. Aquí se tuvieron en cuenta las observaciones del usuario sobre la apariencia de los formularios, haciendo énfasis en que debía ser sencilla y fácil de manejar.

4.4.3 Prototipo de inserción de datos

En este prototipo se implemento el modulo de contratos. Aquí, el usuario escoge la actividad a registrar, la categoría del producto y el producto involucrado, una vez escogido éste, puede ver el costo de la actividad correspondiente a ese producto el cual es el precio que se debe pagar por unidad; luego se digita la cantidad de producto y se escoge de una lista el empleado al cual será asignada la tarea realizada. Hecho esto queda registrado en un contrato previamente creado, la actividad hecha, el producto y el valor que será asignado al empleado para sumar a su pago al final de la semana. Si la actividad realizada fue de 'fundición de producto' se hará una entrada al inventario por la cantidad digitada previamente y así mismo se actualizará en la tabla 'Productos' la cantidad en bodega o stock.

Aquí se cambiaron algunas interfaces del prototipo anterior usando herramientas .NET para inserción de datos a la base de datos del sistema, y procedimientos de consulta.

4.4.4 Prototipo de análisis de información

Las funciones de procesamiento incluye entradas, cálculos, recuperar información y datos de salida en reportes imprimibles. Como los datos muchas veces no son ingresados de la forma correcta o en la secuencia válida, la

aplicación se diseña para asegurar la detección de errores. El objetivo de este prototipo es suministrar al usuario seguridad en cada uno de los procesos y actividades que realiza el sistema; se evalúan procedimientos mediante la implementación de control de errores para los individuos que emplean el prototipo, por medio de manejadores de excepciones o la habilitación y uso de los controles. Además se incluye la opción de realizar copia de seguridad de la base de datos, así como la inclusión de los casos de uso de informes y consultas que permiten la verificación de datos y a partir de esto tomar decisiones por ejemplo sobre los pedidos, la producción o cartera morosa que en el momento de realizar consultas o informes existan. Este es el prototipo final el cual ya el usuario ha trabajado con el prototipo anterior y ha evaluado sus características y operación.

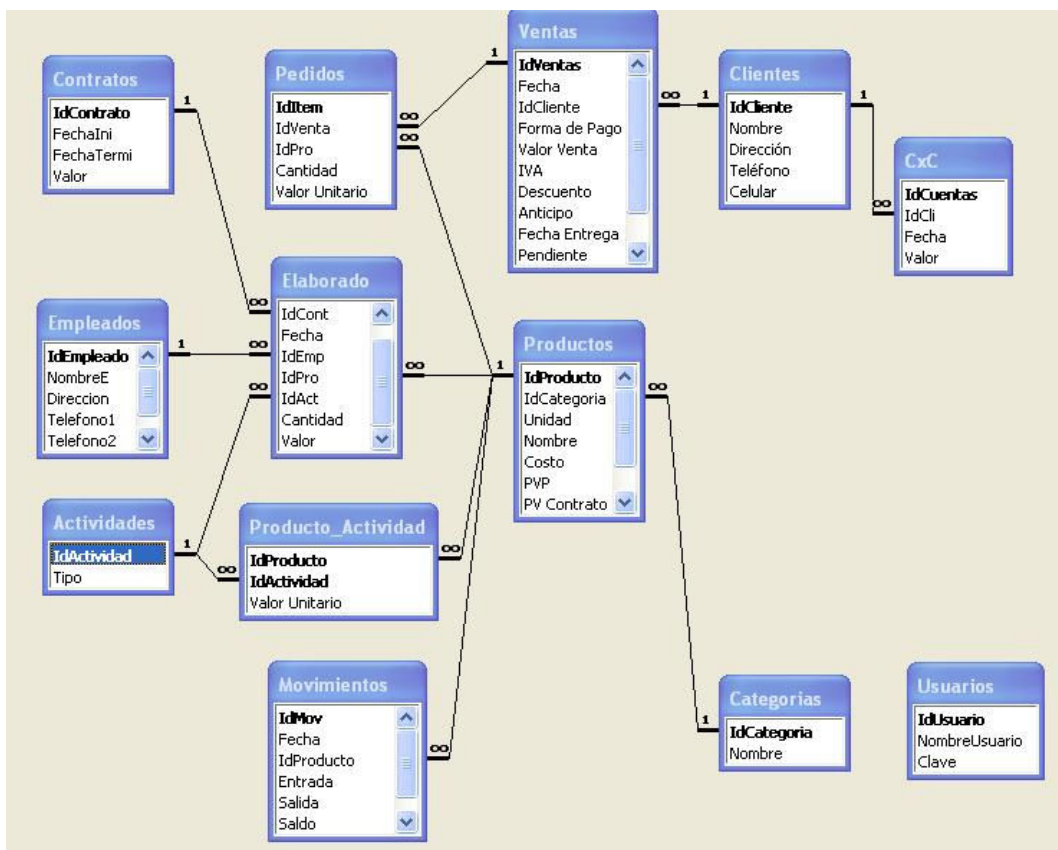
5 IMPLEMENTACION

El modelo de prototipado evolutivo trae implícito una etapa de implementación basada en la revisión, innovación, sugerencias, reacciones iniciales y corrección de cada uno de los prototipos desarrollados. Ésta etapa se inicia con los resultados obtenidos en la etapa de diseño y guía las operaciones y funciones que debe realizar el software mediante código.

5.1 MODELO DE DATOS DEL PROYECTO

La organización de datos es necesaria en el desarrollo del proyecto, la construcción de un modelo de datos asociado al manejo de la aplicación donde se almacenan los datos que controlan su funcionamiento se realiza de acuerdo a los parámetros requeridos. El modelo de datos detallado se encuentra en el Anexo A

Figura 17. Modelo de tablas relacionales



5.2 ESPECIFICACIÓN DE LA INTERFAZ

Esta etapa define el aspecto externo de la aplicación y la manera como van a interactuar con este.

5.2.1 Acceso al programa

Para ingresar al programa se debe dar doble clic en el ejecutable “DIPRECO.EXE” que se encuentra ubicado en la carpeta “C:\Archivos de Programa\DIPRECO”.

Figura 18. Acceso al programa



5.2.2 Inicio de sesión

Después de dar clic en el ejecutable se abre una ventana de autenticación de usuario.

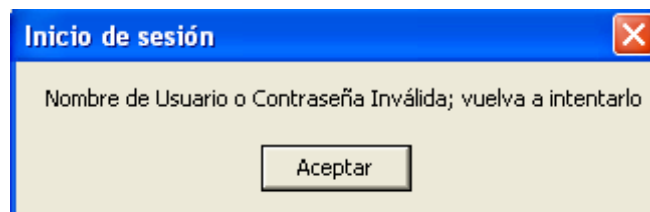
Los datos requeridos para este proceso son Usuario y Clave. Esta información pasa por un proceso de autenticación para evitar que personas ajenas o sin permiso entre al programa.

Figura 19. Inicio de Sesión



Si el usuario no existe o la clave es incorrecta aparece un mensaje advirtiendo de esta situación.

Figura 20. Usuario o Contraseña Inválida



- **Menú Opciones**

En el formulario de inicio de sesión se encuentra el menú Acciones donde el usuario puede cambiar su clave.

Figura 21. Cambiar Clave de Usuario



Una vez digitado el nombre de usuario y la respectiva contraseña se da clic en el botón Aceptar para validar la información suministrada. Si la información es correcta se visualizaran las casillas para efectuar el cambio de contraseña.

Figura 22. Cambiar Clave de Usuario Comprobación correcta



En las casillas se piden la nueva contraseña, luego se debe dar clic en el botón Guardar Cambios para hacer efectivo el cambio de clave. Una vez se valida el usuario y la clave se puede entrar al programa.

5.2.3 Ventana de Menú Principal

Cuando el usuario que entra al programa, se abre una forma con enlaces a los 4 submódulos principales.

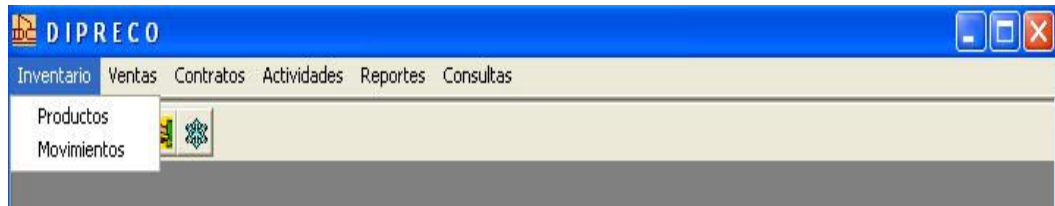
Figura 23. Menú Principal



5.2.4 Submódulo Inventario

En el modulo inventario encontramos 2 secciones: Productos y Movimientos

Figura 24. Menú Inventario



▪ **Productos**

En esta sección el usuario tiene la opción de consultar, editar e insertar registros en la tabla productos. El usuario introduce un nombre de producto a buscar, no es necesario escribirlo completo, basta con escribir la primera letra para mostrar todos los registros que coincidan con dicho caracter. Cuando se listan varios productos que coincidan con el criterio de búsqueda, es posible desplazarse entre estos moviéndose a través del datagrid en el que son listados. Los productos aparecen con todos sus detalles, cabe aclarar en este punto que la empresa maneja 3 precios distintos para cada producto que son: Precio de Venta al Publico (P.V.P), Precio de Venta Contrato Civil (P.V.C.Civil) y Precio de Venta Contrato Suministro (P.V.C.Sumin).

Figura 25a. Formulario Productos

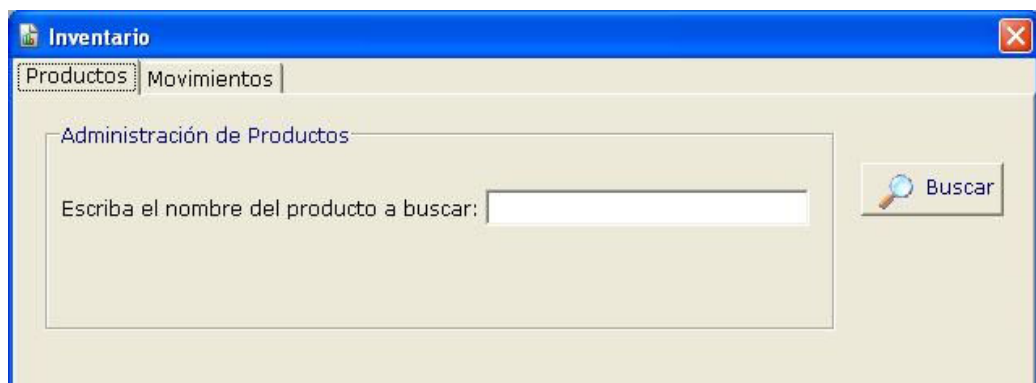
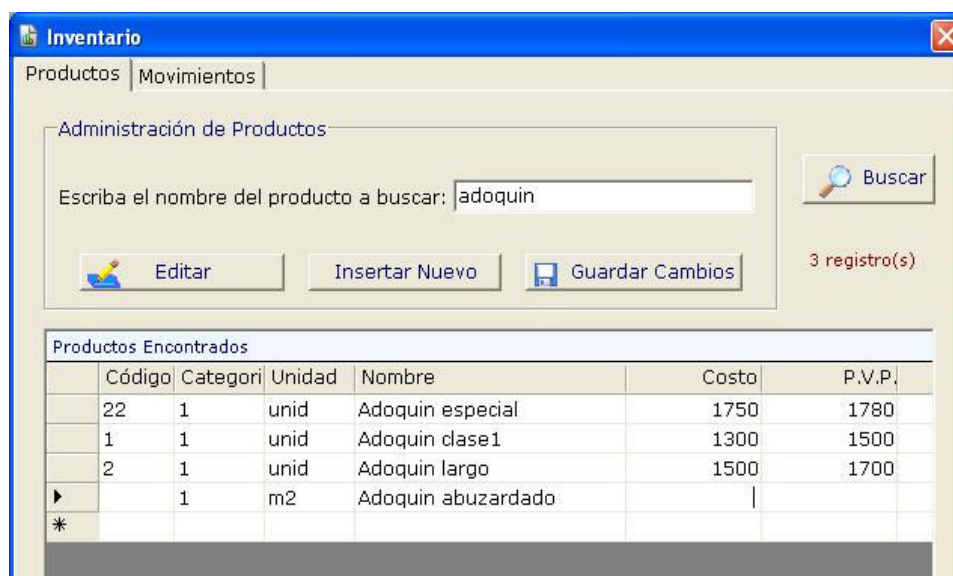


Figura 25b. Formulario Productos



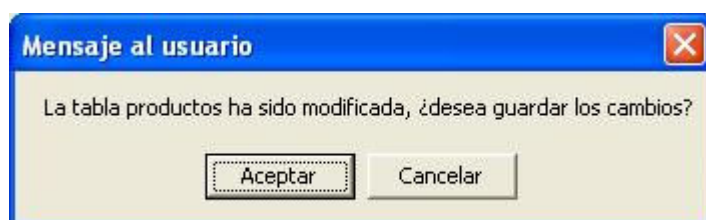
Encontrado el producto si se desean hacer cambios debe hacer clic en el botón 'Editar', el cual habilita al usuario a hacer cambios en todos los campos a excepción del campo 'código' que corresponde al Id del producto y que es inmodificable. También se muestra el botón 'Insertar Nuevo' si lo que se desea es agregar un nuevo registro; si es así, al hacer clic en este botón se habilita la opción de insertar registro, agregando un renglón en blanco a la grilla.

Figura 26. Editar o Insertar Producto



Si el usuario realizó algún cambio y da clic al botón 'Guardar Cambios' aparecerá un mensaje de confirmación.

Figura 27. Guardar Cambios



Si se da clic en 'Aceptar' la tabla Productos es actualizada, si no, la grilla vuelve a ser de solo lectura y los cambios no son guardados.

▪ **Movimientos**

Esta sección es para que el usuario pueda consultar el movimiento del inventario haciendo búsquedas ya sea por fechas o por productos.

Figura 28. Formulario Movimientos



Para hacer consulta por fecha, en la pestaña de este mismo nombre se introducen las fechas entre las que se quiere ver que movimientos se han efectuado en el inventario. A continuación se da clic en el botón 'Ver'

Figura 29. Movimiento de Inventario según la fecha



Relación de Movimientos de Inventario

Fecha | Producto

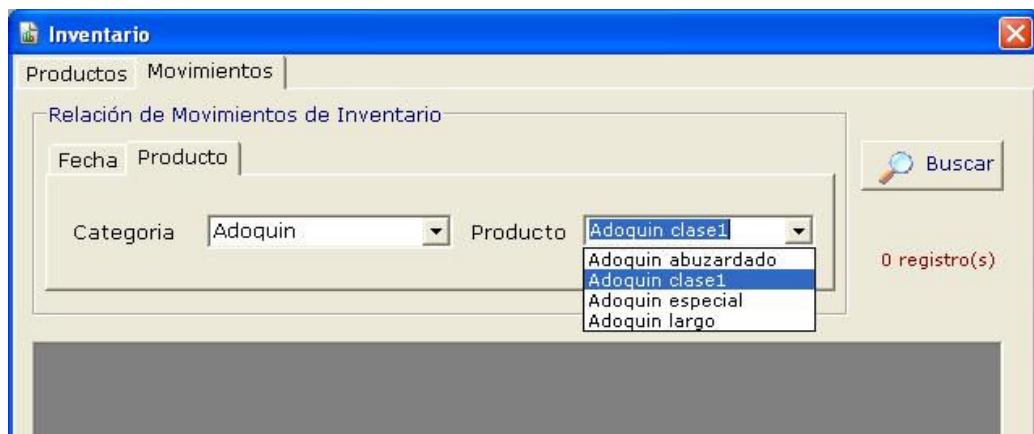
Movimientos de: 01/12/2006 a 01/01/2007

12 registro(s)

	Fecha	Producto	Entrada	Salida	Saldo
▶	04/12/2006	pre1	0	10	50
	04/12/2006	Bloquecito1	0	1	9
	04/12/2006	Bloquecito1	0	5	10
	04/12/2006	Bloquecito1	0	5	15
	04/12/2006	Bloquecito1	0	10	20
	04/12/2006	Adoquin clase1	0	5	60

En la pestaña 'Productos' la consulta se hace escogiendo primero una categoría, luego un producto y se realiza la búsqueda de los movimientos correspondientes al producto escogido y las fechas que se escogieron anteriormente.

Figura 29. Movimiento de Inventario según el producto



Relación de Movimientos de Inventario

Fecha | Producto

Categoría: Adoquin | Producto: Adoquin clase1

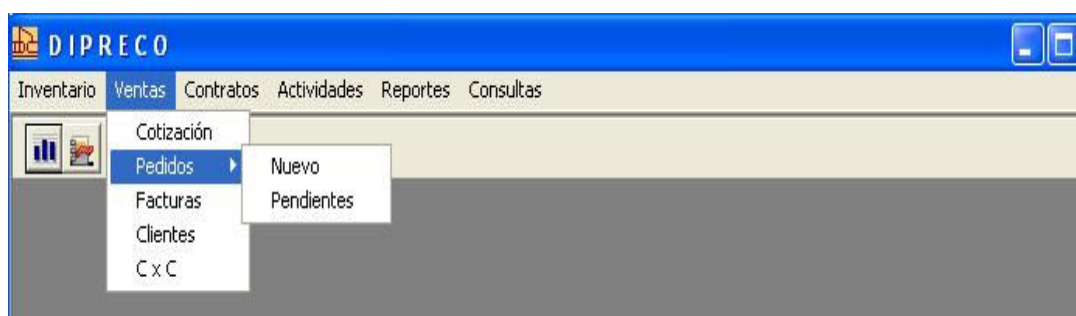
0 registro(s)

	Fecha	Producto	Entrada	Salida	Saldo
--	-------	----------	---------	--------	-------

5.2.5 Submódulo Ventas

En el formulario del módulo ventas se encuentran 5 secciones: Cotización, Pedidos, Facturas, Clientes y Cuentas por Cobrar.

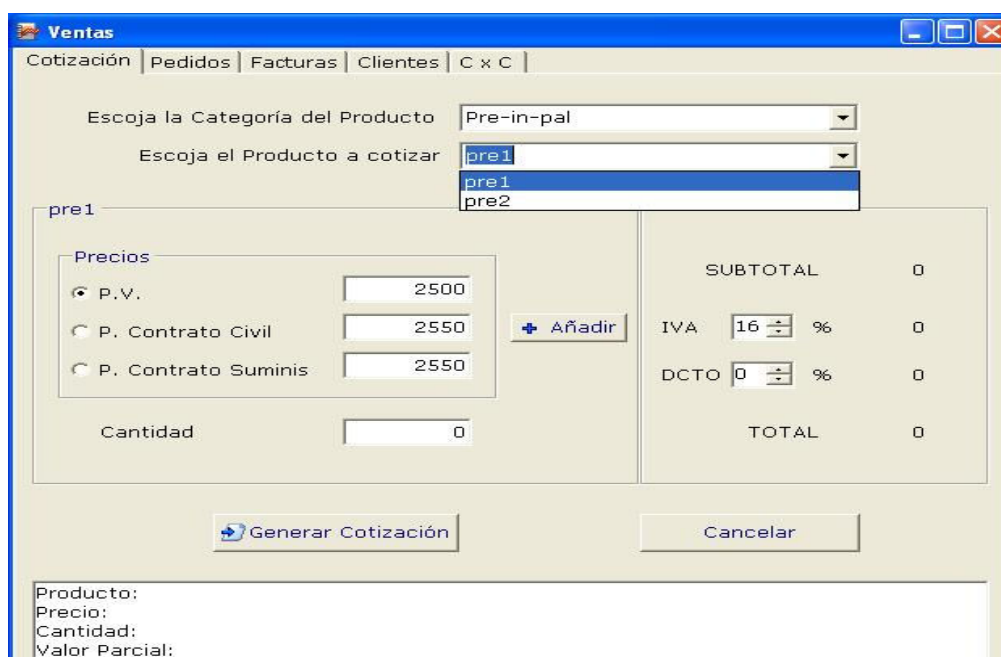
Figura 30. Menú Ventas



▪ **Cotización**

En este formulario se añaden productos a cotizar, se calcula el valor total de la cotización y se genera la misma en un documento de Word. Primero se debe escoger una categoría y un producto, una vez hecho esto aparecen en el lado izquierdo del formulario los 3 precios del producto para q el usuario escoja con cual se va a cotizar; se introduce un valor de cantidad y se da clic en añadir. El producto y los datos de precio, cantidad y valor aparecen en el cuadro de lista inferior.

Figura 31. Formulario de Cotización



SUBTOTAL		0
IVA	16 %	0
DCTO	0 %	0
TOTAL		0

A medida que se van añadiendo productos los cálculos de valores parciales, IVA, y el total se van sumando y van apareciendo en el lado derecho del formulario, como se muestra en la siguiente figura.

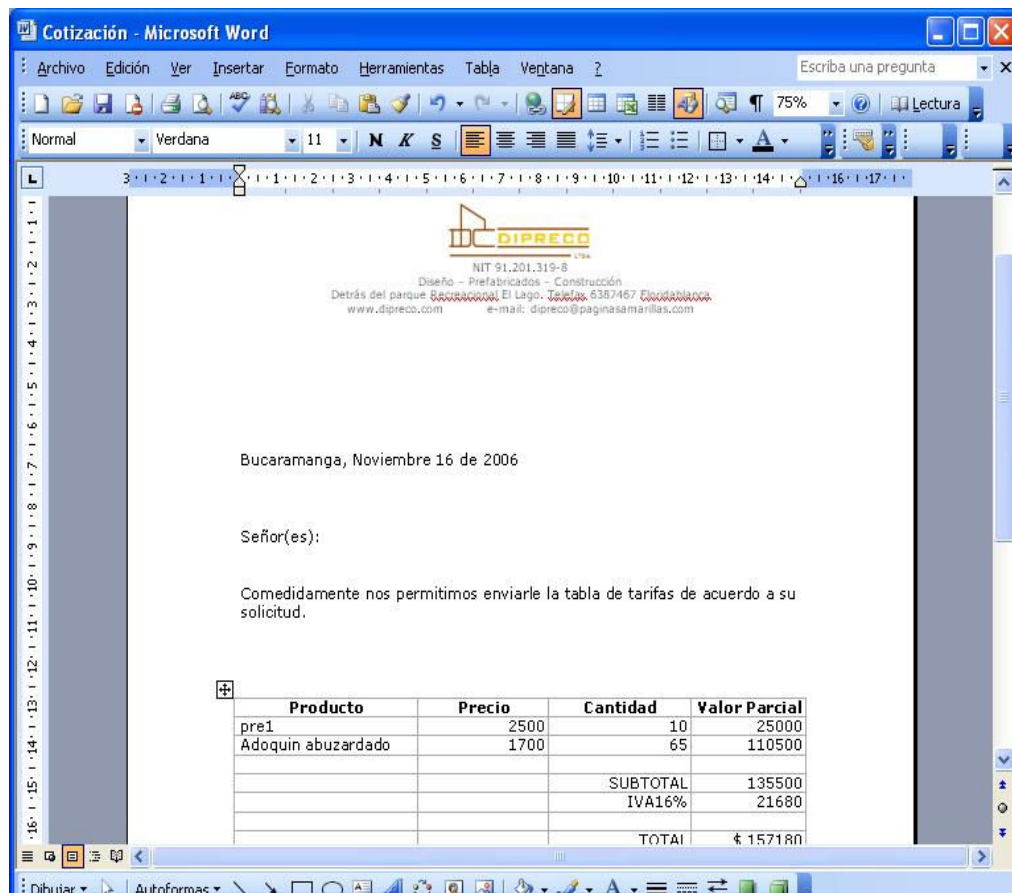
Figura 31b. Formulario de cotización

Adoquin abuzardado	
Precios	
<input type="radio"/> P.V.	1650
<input checked="" type="radio"/> P. Contrato Civil	1700
<input type="radio"/> P. Contrato Suminis	1630
<input type="button" value="+ Añadir"/>	
Cantidad	65
SUBTOTAL 135500	
IVA	16 % 21680
DCTO	0 % 0
TOTAL \$ 157180	

Producto:	pre1	Adoquin abuzardado
Precio:	2500	1700
Cantidad:	10	65
Valor Parcial:	25000	110500

Cuando ya se han añadido todos los productos a cotizar, se da clic en 'Generar Cotización' y se abrirá un documento de Word con el formato básico de cotización en el que se ha generado una tabla de los productos añadidos, con sus cantidades, precios y valores. El usuario puede hacer modificaciones si lo considera, e imprimir. Si se da clic al botón 'Cancelar' los datos se borrar y el formulario queda en blanco.

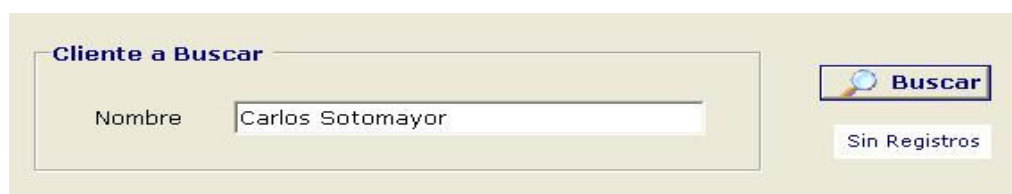
Figura 31. Documento generado de Cotización



▪ **Pedidos**

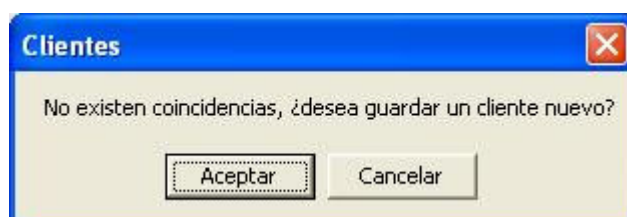
En este formulario se registran los pedidos. El pedido es asignado a un cliente que bien puede estar en la base de datos o ser un cliente nuevo en cuyo caso se solicitaran los datos para registrarlo como cliente nuevo. Para saber si el cliente ya existe se hace una búsqueda por nombre

Figura 33. Formulario Pedidos, búsqueda de cliente por nombre



Si el cliente no existe se mostrará un mensaje de alerta

Figura 34. Mensaje de alerta



Si se da clic en 'Aceptar' el formulario habilita los campos para que sean llenados con los datos del nuevo cliente y se muestra el botón 'Guardar'.

Figura 35. Guardar Cliente Nuevo

A screenshot of a web application interface. At the top, there is a search section titled 'Cliente a Buscar'. It contains a text input field with 'Nombre' and the value 'Carlos Sotomayor'. To the right is a 'Buscar' button with a magnifying glass icon. Below the search input is a 'Sin Registros' message. In the center, there is a 'Guardar' button with a floppy disk icon. Below that is a form for saving a new client. It has a 'Nombre' field with 'Carlos Sotomayor'. Below it are three fields: 'Dirección', 'Teléfono', and 'Celular', each with an empty input box.

Si por el contrario, el cliente existe, se muestran las coincidencias encontradas y los datos del cliente respectivo, si hay varios clientes con el mismo nombre el usuario puede escoger de la lista el cliente cuyos datos se ajustan a su búsqueda.

Figura 36. Búsqueda de cliente, cliente encontrado

Cliente a Buscar

Nombre

1 de 1

Cientes Encontrados: Código

Nombre

Dirección	Teléfono	Celular
la victoria	321458	125479965

Una vez se tiene el cliente se pasa a escoger de una lista los productos que serán pedidos; se escoge el tipo de precio y este automáticamente se carga de la base de datos. Se escoge valor de IVA, se digita cantidad y se van añadiendo a una lista como se puede ver en la siguiente figura. Si se desea quitar algún producto de la lista, se marca el checkbox correspondiente y se da clic en el botón 'Quitar'

Figura 37. Formulario de Pedidos, añadir productos

Productos

Categoría Cantidad

Producto IVA %

Precio Valor \$

Lista (2)

Cod	Producto	Precio	Cant	Parcial	IVA	<input type="checkbox"/>
1	Adoquin clase1	1500	10	15000	2400	<input type="checkbox"/>
20	pre1	1350	80	108000	17280	<input type="checkbox"/>

A medida que se van añadiendo productos a la lista, los valores de venta y del IVA van siendo calculados. Cuando ya se han agregado todos los productos el usuario debe escoger las fechas de pedido y de entrega del mismo, así como la forma de pago y el porcentaje de descuento si lo hubiere. El usuario puede, si es pertinente, una cantidad correspondiente a un anticipo dado por el cliente. Al dar clic en el boton 'calcular' se muestra el valor final a pagar o el saldo pendiente de la venta.

Figura 38. Formulario Pedidos, datos de venta

Fecha Pedido	01/01/2007	Valor Venta	264750	
Fecha Entrega	05/01/2007	IVA %	16	42360
Forma de Pago	Crédito	Dcto %	0	0
		Valor Total	307110	
		Anticipo	0	
		Saldo Pendiente	307110	

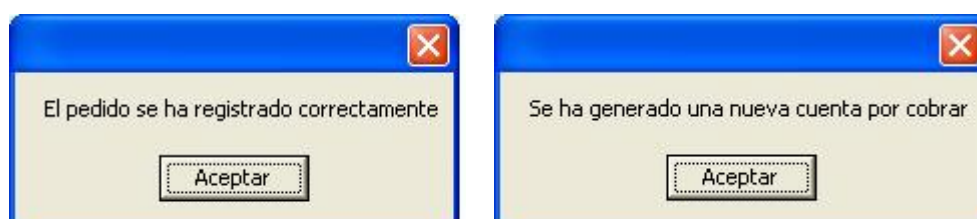
Calcular

En este punto el pedido está listo para ser registrado, finalmente el usuario da clic al botón 'registrar pedido' y si todos los datos están correctos se creará un nuevo pedido y se actualizará el inventario si es un pedido de entrega inmediata, si no, se crea un pedido pendiente. Si la venta fue a crédito se creará una cuenta por cobrar a cargo del cliente y se muestra un aviso.

Figura 39a. Botón registrar, formulario pedidos



Figura 39b. Mensajes de confirmación



Cuando el pedido está para entregar en una fecha posterior a la fecha en la que fue realizado, el pedido es clasificado como pendiente y estará así hasta que sea entregado en la fecha que corresponde. La descarga en el inventario no se hará sino hasta que esto suceda. Para saber que pedidos están pendientes se hace clic en la sección 'Pendientes' del formulario Pedidos. Aquí, se muestra una lista de los números de venta que tienen esta condición. Al seleccionar un número se muestra el nombre del cliente que hizo el pedido, y una tabla con los productos por entregar con sus cantidades, como se ve en la figura

Figura 40. Formulario pedidos pendientes

28/01/2007 **Pedidos con Entregas Pendientes**

Venta No

63
75
76
77
80
99

➔

Fecha Entrega 05/02/2007

Cliente Rocio Perez

Dirección calle 18 No 31-25

Productos a Entregar			
Código	Producto	z	Cantidad
20	pre1		150

Actualizar

Registrar Entrega

Para registrar la entrega del producto, se hace clic en el botón 'Registrar Entrega'. En ese momento el pedido deja de estar clasificado como 'pendiente' y se hace la descarga en el inventario de los productos que salieron.

En esta misma sección, el usuario registra las salidas de inventario por reposición, es decir, cuando se presentan rupturas o desperdicio de productos en el cargue o descargue al realizar las entregas, estas salidas por concepto de reposición pueden ser consultadas después si se quiere sacar un balance de las mismas. Cuando esta salida se registra, se actualiza el inventario.

Figura 41. Salidas de productos por reposición

Reposiciones

Categoría Adoquin

Producto Adoquin especial

Cantidad

Actualizar Inventario

▪ Facturas

En este formulario, el usuario puede cargar los datos completos de un pedido en formato de factura escogiendo un cliente de una lista, y un número de venta correspondiente al mismo. Al hacer clic en el botón 'Cargar Factura' el formulario se llena con todos los datos.

Figura 42. Formulario facturas

Busqueda por Cliente

Cientes Registrados

Ventas Correspondientes al Cliente (Codigo de Venta)

[Cargar Factura](#)

[Generar](#)

Fecha: 18/10/2006 **Factura No** 0872

Señor: Andrea Monroy

Dirección: cra 30 No 17-22 San alonso

Teléfono: 6321458 **Forma de pago:** Crédito

Unidad	Descripción	Cantidad	Valor Unitario	Total
unid	Bloque2	150	1600	240000
M2	pre1	200	2550	510000

Pagado

Pendiente

Subtotal	750000
IVA	16% 120000
Descuento	5% 43500
TOTAL	826500

El usuario puede generar la factura en un documento de Word si la quiere imprimir, dando clic al botón 'generar'.

▪ Cuentas por Cobrar

Este formulario muestra un listado de las cuentas pendientes

Figura 45. Formulario cuentas por cobrar

Listado de Cuentas por Cobrar					
	Fecha	Cuenta	Cliente	Nombre	Valor
	18/10/2006	4	26	hilda perez	1958870
▶	20/10/2006	10	25	carolina mejia	17226
				<input type="button" value="Actualizar"/>	
				TOTAL	\$ 1.976.096,00

Para consultar las cuentas pendientes de un cliente en particular, se selecciona el registro con el cursor; el código del cliente aparece en una caja de texto, el usuario da clic a botón 'consultar' y se carga una tabla con las deudas pendientes de ese cliente en particular.

Figura 46. Consultar CxC por Cliente

Listado de Cuentas por Cobrar					
	Fecha	Cuenta	Cliente	Nombre	Valor
▶	18/10/2006	4	26	hilda perez	1958870
	20/10/2006	10	25	carolina mejia	17226
				<input type="button" value="Actualizar"/>	
				TOTAL	\$ 1.976.096,00
Consulta por Cliente					
			Cliente Seleccionado	<input type="text" value="26"/>	<input type="button" value="Consultar"/>

Para registrar el pago de una deuda, se selecciona el registro con el cursor, el código de la cuenta aparece en una caja de texto y se da clic al botón 'Pagado'

Figura 47. Registrar pago de CxC

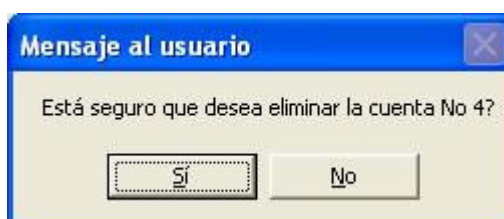


Registrar Pago

Cuenta Seleccionada 4

Después de dar clic al boto 'pagado' se muestra un mensaje de confirmación

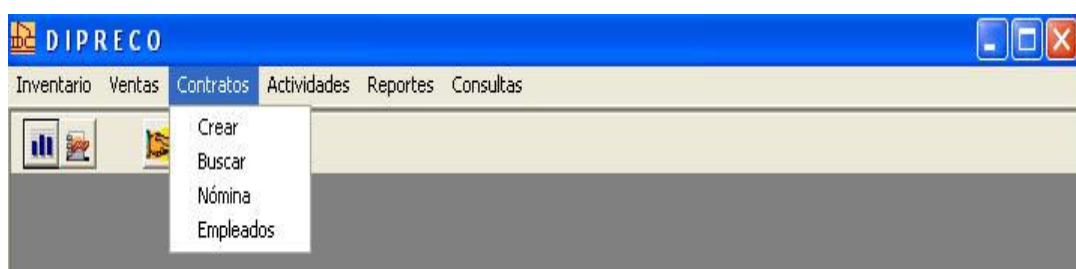
Figura 48. Mensaje de confirmación de pago CxC



5.2.6 Submódulo Contratos

En este submódulo encontramos las secciones: crear, buscar, nómina y empleados.

Figura 49. Menú Contratos



- **Crear**

En la sección crear, como su nombre lo dice, se crean nuevos contratos. Se escogen fechas de inicio y terminación y se da clic al botón crear.

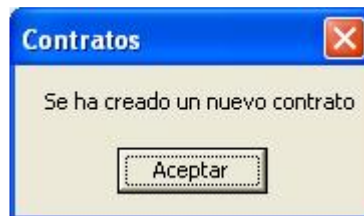
Figura 50. Formulario crear contrato



The screenshot shows a web application window titled "Contratos". At the top, there is a navigation menu with tabs: "Crear", "Buscar", "Nómina", and "Empleados". The "Crear" tab is active. Below the menu, there is a form titled "Contrato Nuevo". The form contains two date selection fields: "Fecha Inicio" and "Fecha Terminación". Both fields are currently set to "01/01/2007". Below these fields is a "Crear" button with a small icon to its left.

Se muestra un mensaje de confirmación al usuario después de crear el contrato

Figura 51. Mensaje de confirmación, crear contrato



The screenshot shows a small dialog box titled "Contratos" with a close button (X) in the top right corner. The main text of the dialog is "Se ha creado un nuevo contrato". Below the text is a single button labeled "Aceptar".

Una vez el contrato es creado se muestran el numero de contrato en estado activo y se habilita un enlace para empezar a agregar actividades realizadas al mismo.

Figura 52. Formulario contratos



The screenshot shows the "Contratos" application window after a contract has been created. The "Contrato Nuevo" form is still visible, but the "Fecha Terminación" field is now set to "06/01/2007". The "Crear" button is now disabled. Below the form, there is a section showing the contract details: "Contrato No" followed by a text input field containing the number "8", and "Estado: Activo". At the bottom of the window, there is a blue link labeled "Agregar Actividades".

- **Buscar**

En esta sección se busca el contrato que se quiere ver en una lista y se da clic en el botón 'ver detalles'. Se muestran los datos y los detalles de actividades realizadas en este contrato.

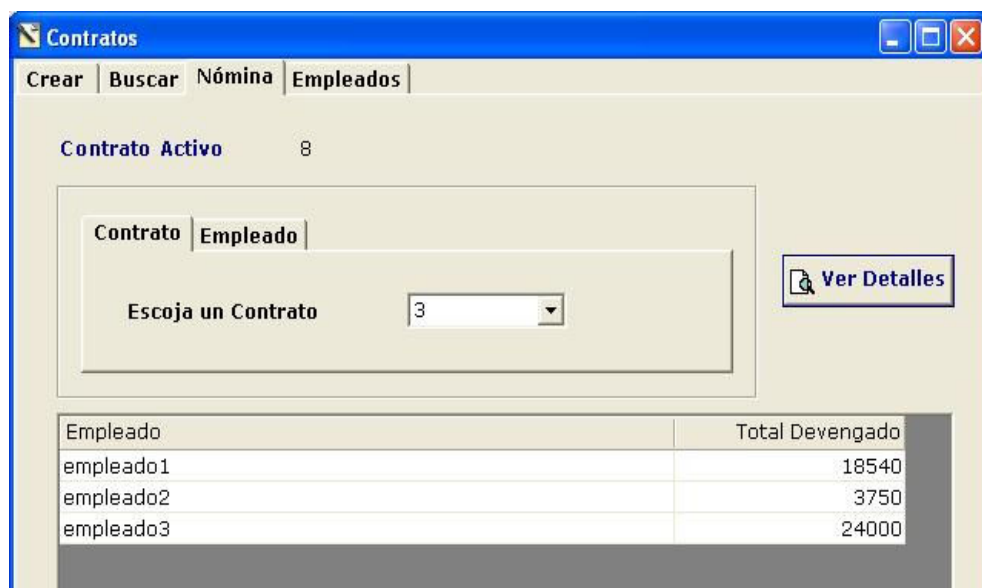
Figura 53. Buscar contrato



- **Nómina**

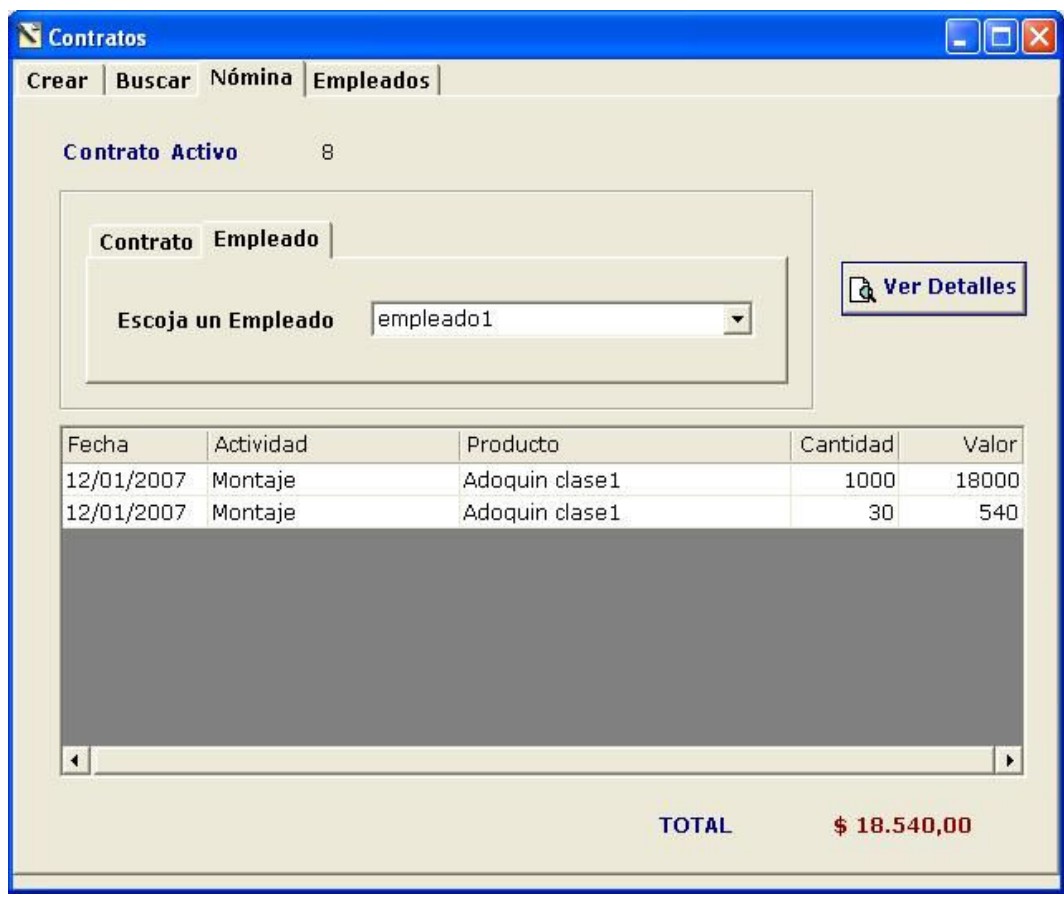
En esta sección se puede consultar el valor de la nómina a pagar. Las consultas se pueden hacer por contrato o por empleado. La consulta por contrato se hace escogiendo de una lista el número de contrato del que se desea conocer el valor total de la nómina a pagar.

Figura 54. Nómina, consulta por contrato



En la consulta por empleado, se escoge de una lista el empleado del que se desea conocer los detalles de su total devengado, mostrando las actividades que dicho empleado realizó durante la duración del contrato.

Figura 55. Nómina, consulta por empleado



▪ **Empleados**

En esta sección se administran los datos de los empleados con las funciones de consulta, edición, inserción y eliminación de registros.

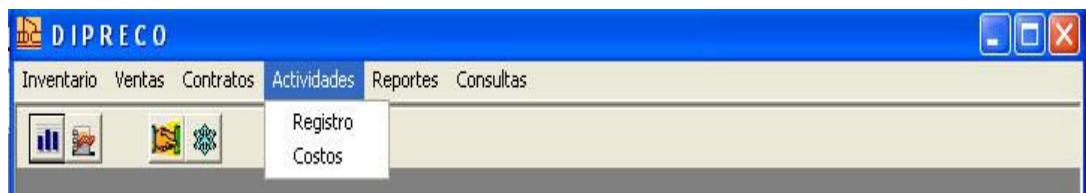
Figura 56. Formulario Empleados



5.2.7 Submódulo Actividades

En este submódulo se llevan a cabo las tareas de registro de actividades diarias realizadas en la planta de producción. Aquí encontramos las secciones Registro y Costos.

Figura 57. Menú Actividades



- **Registro**

En este formulario se hace el registro de actividades escogiendo de una lista el tipo de actividad, el producto, se digita la cantidad de producto y se escoge el empleado que la desempeñó y al cual será asignada. Al dar clic al botón 'Añadir a contrato' la actividad queda registrada en el contrato que se encuentre activo en ese momento

Figura 58. Formulario Actividades, registro

▪ **Costos**

Aquí se pueden consultar los valores de los costos por actividad. Este valor es el que es pagado al empleado por unidad de producto cuando realiza dicha actividad. Se escoge el producto del que se quieren saber los detalles de costos según la actividad. Estos costos pueden ser modificados.

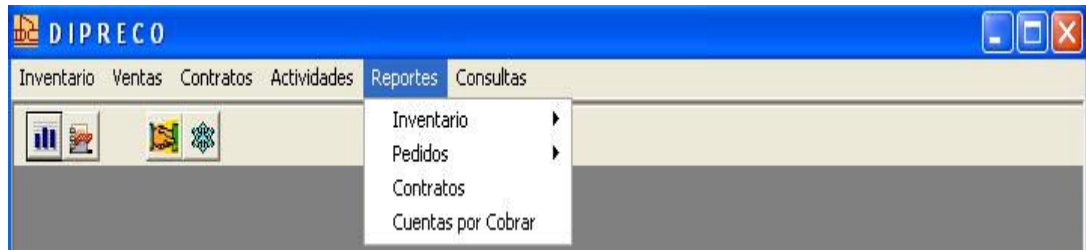
Figura 59. Formulario Actividades, costos

COSTOS	
Actividad	Costo
Fundida	150
Montaje	18
Cargue y descargue	100

5.2.8 Reportes

Se implementaron reportes que pueden ser consultados e impresos por el usuario. El usuario tiene acceso a reportes de: Movimientos de Inventario, ventas, Pedidos (historial y pendientes), contratos y nómina, y por último, cuentas por cobrar.

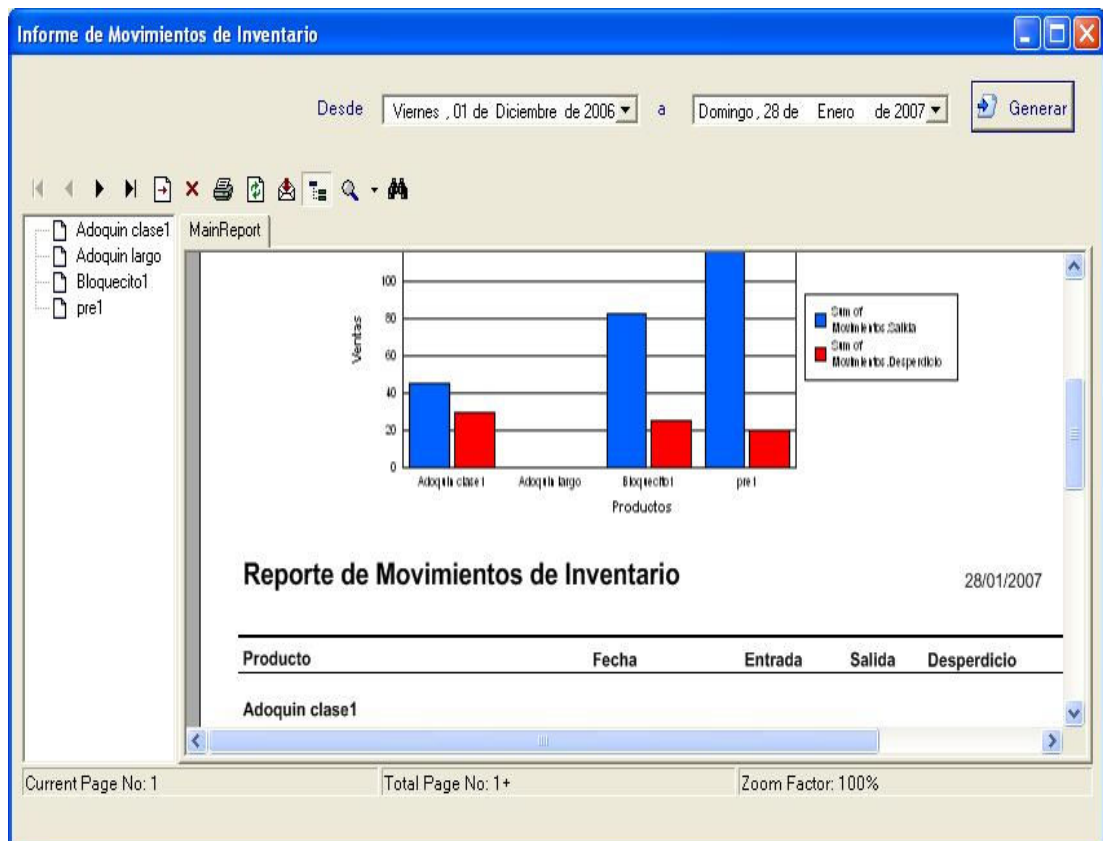
Figura 60. Menú Reportes



- **Reporte Inventario**

El usuario escoge las fechas en las que desea ver el movimiento del inventario, el reporte muestra las entradas, salidas, y el desperdicio por producto. Una gráfica muestra las salidas de producto, por concepto de ventas y por concepto de desperdicio.

Figura 61. Reporte inventario



▪ Reporte Pedidos

Se generan reportes del historial de pedidos y de los pedidos pendientes de entrega. El usuario escoge las fechas entre las que desea conocer los pedidos que se realizaron en ese intervalo. También se pueden conocer las ventas hechas por producto según las cantidades, dando clic en el botón 'Ventas'.

Figura 62a. Reporte Pedidos

Reporte de Pedidos

Reporte de Pedidos entre 01/12/2006 Y 28/01/2007 Generar

VENTAS

95
96
97

MainReport

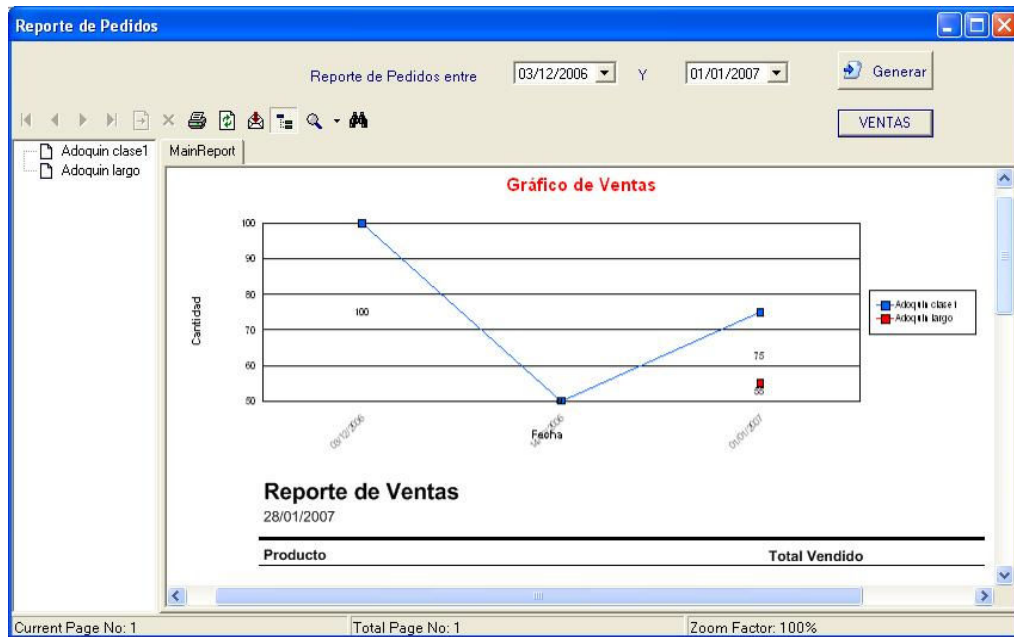
IDC DIPRECO LTDA.

Reporte de Pedidos 28/01/2007

Venta No	Cliente	Producto(s)	Cantidad
95	Nubia Blanco		
	03/12/2006	Adoquin clase1	100

Current Page No: 1 Total Page No: 1 Zoom Factor: 100%

Figura 62b. Reporte Pedidos, ventas



▪ **Reporte Contratos**

Se escoge de una lista el número de contrato del que se quiere generar el reporte. Se puede ver un reporte de nómina con los detalles de lo devengado por cada empleado al dar clic en el botón 'Nomina'

Figura 63. Reporte de contrato

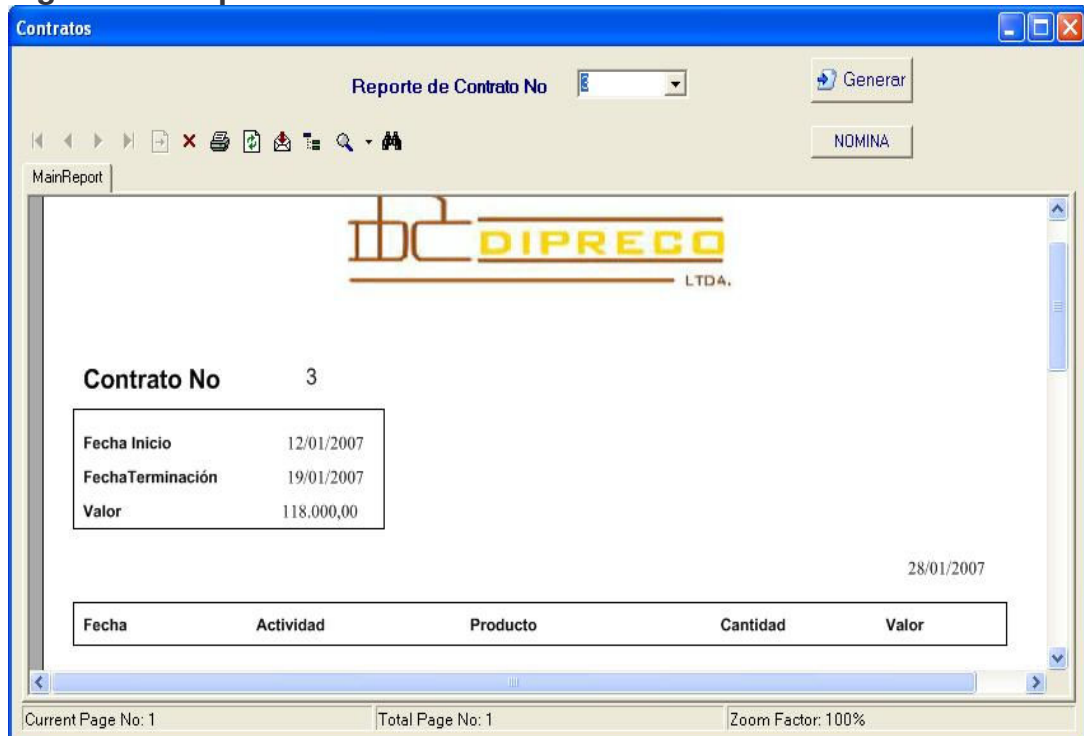


Figura 63. Reporte de nómina de pago a destajo

Contratos

Reporte de Contrato No 3 Generar

NOMINA

MainReport

Nómina

Contrato No	3
Fecha Inicio	12/01/2007
Fecha Terminación	19/01/2007
Valor	118.000,00

Empleado	Actividad	Producto	Cantidad	Valor
empleado1	Montaje	Adoquin clase I	30	540
	Montaje	Adoquin clase I	1.000	18.000

Current Page No: 1 Total Page No: 1 Zoom Factor: 100%

▪ **Reporte Cuentas por Cobrar**

Se muestra un listado de las cuentas por cobrar, con sus respectivos detalles

Figura 64. Reporte de cuentas por cobrar

<u>Fecha</u>	<u>Nombre</u>	<u>Valor</u>
<u>18/10/2006</u>	hilda perez	1.958.870,00
		<u>1.958.870,00</u>
<u>20/10/2006</u>	carolina mejia	17.226,00

5.2.9 Consultas

El usuario puede generar listados de productos, empleados y clientes con sus detalles respectivos, en formato de reporte que pueden ser impresos.

Figura 65. Menú consultas



Figura 66. Listado productos

LISTADO DE PRODUCTOS 29/01/2007

<u>Categoría</u>	<u>Unidad</u>	<u>Producto</u>	<u>Costo</u>	<u>PVP</u>	<u>PV Cont. Civil</u>	<u>PV Cont. Suministro</u>	<u>Cantidad Bodega</u>
Adoquin							
	unid	Adoquin clase1	1.300	1.500,00	1.560,00	1.600,00	105
	m2	Adoquin corto	1.600	1.650,00	1.700,00	1.630,00	120
	unid	Adoquin especial	1.750	1.780,00	1.740,00	1.765,00	85
	unid	Adoquin largo	1.500	1.700,00	1.750,00	1.720,00	165
Bloque							
	unid	Bloque2	1.400	1.500,00	1.582,00	1.600,00	90
	unid	bloque3	1.320	1.600,00	1.650,00	1.700,00	80
	unid	Bloquecito1	1.255	1.800,00	2.000,00	2.550,00	7
Pre-in-pal							

6 PRUEBAS Y REVISION DE PROTOTIPOS

En esta etapa se explica el método iterativo y las responsabilidades a los usuarios en la evaluación de la aplicación ya que el usuario participa directamente en todo el proceso. Es tarea del usuario trabajar con el prototipo y evaluar sus características y funcionamiento; la experiencia con el sistema bajo condiciones reales permite la familiaridad indispensable para determinar los cambios o mejoras que sean necesarios.

Esta información tiene influencia en las posteriores versiones del prototipo, la cuales se presentan modificadas o refinadas. Las etapas descritas anteriormente se repiten varias veces hasta que usuarios y desarrolladores estén de acuerdo en que el prototipo ha evolucionado lo suficiente o que una iteración más no traerá beneficios adicionales. Durante el desarrollo de cada prototipo de software se llevaron a cabo dos tipos de pruebas: pruebas en tiempo de desarrollo y pruebas después de la programación.

El orden de las pruebas se llevo a cabo de la siguiente forma:

- **Prototipo básico de requerimientos.** Se lleva a cabo un proceso de inserción de datos con los cuales se realizan procesos de registro de pedidos, y actividades con la respectiva actualización del inventario.
- **Prototipo de interfaces.** El usuario del sistema evalúa la distribución y organización del espacio visual de la pantalla así como la facilidad de manejo.
- **Prototipo de inserción de datos.** Se comprueba la actualización de los registros relacionados en la base de datos al insertar, modificar o eliminar registros en los diferentes formularios
- **Prototipo análisis de información.** Se generar informes, consultas y se verifica el correcto manejo de la información existente.
- **Prueba Final.** Esta prueba realiza una detallada inspección en el sistema con el fin de asegurar un óptimo funcionamiento de la herramienta completa.

Las pruebas en tiempo de desarrollo fueron realizadas durante toda la etapa de programación de los módulos que conforman el software.

Las pruebas para los prototipos se efectúan cuando se considera terminado el prototipo. El desarrollador realizó pruebas para verificar que el manejo de datos existentes fuese verídico, garantizar la correcta realización de acciones sugeridas por el usuario y validar datos extraídos de la base de datos existente. El usuario final realizó pruebas que ayudaron a detectar fallas en la interfaz de usuario, fallas en la apariencia de los menús, funciones incorrectas o ausentes, errores en estructuras de datos, errores de inicialización y terminación, errores en la comprensión de procedimientos y para detectar innovación en el software.

Para las pruebas se utilizaron los siguientes recursos:

Tabla 3. Recursos de las pruebas

Recursos	
Hardware	Procesador Pentium IV a 2.4 Ghz Memoria RAM de 512 MB Disco duro de 40 GB Monitor de 17'
Software	Sistema Operativo Microsoft Windows XP Server. Motor de base de datos Microsoft OLEDB Driver 4.0 (Access 2000) Explorer 6.0

El último prototipo se probó en diferentes equipos con características hardware y software diferentes, con el fin de verificar el correcto funcionamiento en equipos con mantenimiento adecuado fuera del alcance de implantación del software desarrollado.

6.1 PROTOTIPO BASICO DE REQUERIMIENTOS

6.1.1 Descripción de la prueba

El objeto de esta prueba era determinar los posibles errores en la comprensión y manejo de procesos para los requerimientos básicos en el desarrollo del software, en los que se tuvieron en cuenta los siguientes aspectos:

- Rutinas básicas para el registro de pedidos, cálculo de valores de venta, IVA, descuentos, etc.
- Datos de entrada necesarios para registrar producción y movimientos en el inventario
- Manejo y aclaración de variables de sesión.
- Datos necesarios de la base de datos para generar facturas

6.1.2 Resultados de la prueba

Las pruebas de este prototipo fueron elaboradas por el desarrollador. Como resultado de la prueba se obtuvieron las siguientes observaciones a tener en cuenta:

- Se presentaron discordancias en los datos del campo 'stock' en la tabla Productos y los datos del movimiento diario de inventario por problemas en la rutina de actualización del mismo al registrar un pedido.

- Se aclararon los datos necesarios para el registro de actividades diarias en la planta de producción, y otros conceptos procedentes de la base de datos y aquellos insertados por el usuario del sistema.
- Se tomó como referencia de interfaz de diseño para el formularios de generación de factura de pedido, los datos requeridos de la base de datos y los usados en la factura actual en la empresa.

Los errores y observaciones mencionados fueron corregidos e implementados en el desarrollo del segundo prototipo, al cual se realizaron pruebas informales antes de darlo por terminado.

6.2 PROTOTIPO DE INTERFACES

6.2.1 Descripción de la prueba

El objeto de esta prueba fue determinar el nivel de satisfacción de los usuarios en cuanto a colores, distribución y organización del espacio visual de la pantalla y descubrir innovaciones en el software de acuerdo a las necesidades de cada usuario.

Particularmente se tuvieron en cuenta los siguientes aspectos:

- Inicio de sesión y permisos permitidos según el tipo de usuario.
- Seguimiento a las observaciones del prototipo anterior.
- Validación de los datos a ingresar en la base de datos
- Verificación de enlaces para las opciones de cada usuario.
- Validación de formularios según su estado.
- Combinación de colores e iconos para controles.
- Distribución de espacio en cada formulario.
- Diseño de interfaz acorde a las tareas del usuario.
- Verificar la prohibición del acceso a usuarios no registrados.

6.2.2 Resultados de la prueba

Las pruebas para éste prototipo fueron desarrolladas por el usuario final.

- Las observaciones hechas en el modulo anterior fueron implementadas correctamente al igual que los errores corregidos.
- El ingreso de usuarios presenta errores, al tratar de abrir el menú principal después de validar los datos.
- Los colores en el diseño general no son los más adecuados, y el tamaño de letra es muy pequeño.
- Algunos nombres de los botones no son acordes al lenguaje usado por los usuarios.
- Algunos formularios no están validando el estado de activo, permitiendo múltiples accesos a dicho formulario.
- Algunos informes están siendo omitidos.

Los errores mencionados fueron corregidos en su totalidad y las recomendaciones acatadas en la implementación del tercer prototipo y

evaluados en una verificación que se llevó de manera informal antes de la prueba del tercer prototipo.

6.3 PROTOTIPO DE INSERCIÓN DE DATOS

6.3.1 Descripción de la prueba

El objetivo de esta prueba fue verificar la inserción de datos a la base de datos del sistema, y procedimientos en la consulta y generación de factura de pedido, con información ya almacenada; en los que se tuvieron en cuenta los siguientes aspectos:

- Conexión a la base de datos
- Ingreso de registros.
- Ejecución de sentencias SQL.
- Modificación y eliminación de registros
- Correcta ejecución de las consultas SQL (resultados de búsquedas)
- Funcionamiento del control Datagrid en formularios.
- Manejo de variables de sesión.
- Formato de factura imprimible.

6.3.2 Resultado de la prueba

Las pruebas para éste prototipo fueron desarrolladas por el usuario final. Se obtuvieron las siguientes recomendaciones:

- Las observaciones hechas al prototipo anterior fueron implementadas, tales como Interfaz gráfica y validación de formularios.
- La conexión a la base de datos correcta en todas las consultas SQL.
- Falta validar algunos campos relacionados con el ingreso de registros.
- La modificación y eliminación en los respectivos formularios es correcta.
- Ejecución de sentencias SQL correctas.
- El control Datagrid en cada formulario donde aparece esta funcionando correctamente.
- El formato de factura imprimible presenta datos faltantes.
- Se observa se hace necesario presentar el registro de actividades y los contratos en formularios diferentes.
- Los datagrids presentan los datos en forma desordenada, se necesitan crear estilos para la forma de mostrar en pantalla los datos, el tamaño y la distribución de las columnas, la alineación de los datos, etc.

6.4 PROTOTIPO PARA ANÁLISIS DE INFORMACIÓN

6.4.1 Descripción de la prueba

El objeto de esta prueba fue verificar el manejo de datos generados en las diferentes actividades mediante informes y procesos generales del registro de pedidos y cuentas por cobrar y de registro de actividades y asignación de estas al empelado correspondiente. En esta prueba se tuvieron en cuenta los siguientes aspectos:

- Funcionamiento de reportes de pedidos y movimientos de inventario y por intervalo de fecha según se requiera.
- Funcionamiento de reportes imprimibles para informes en general.
- Concordancia entre los datos existentes en la base de datos y la información generada a partir de estos.
- Validación de datos ingresados por el usuario del sistema.
- Revisión del adecuado funcionamiento de los cambios realizados en el anterior prototipo.

6.4.2 Resultados de la prueba

Las pruebas fueron realizadas por el usuario final y arrojaron resultados satisfactorios. Se cambiaron algunas formas de insertar datos que se hacían digitando por la selección de dicho dato de una lista. Se corrigieron y validaron algunas rutinas

6.5 PRUEBA FINAL

6.5.1 Descripción de la prueba

El propósito de esta prueba fue revisar todo, en conjunto, el sistema software, para asegurar el correcto funcionamiento del mismo. Para esta prueba se tuvo en cuenta lo siguiente:

- Validación de datos en cajas de texto.
- Conexión a la base de datos.
- Ingreso, edición y eliminación de registros de la base de datos.
- Revisión de rutinas
- Verificación de funcionamiento de informes imprimibles tales como: movimientos de inventario, nómina, informe de cuentas por cobrar y en general todos los tipos de informes que se generan en el sistema.

6.5.2 Resultado de la prueba

Como resultado se obtiene a satisfacción por parte de Dipreco y del desarrollador un sistema software que cumple con los requerimientos del proceso de control de inventarios, cuentas por cobrar, nómina y otros conceptos y el correcto funcionamiento de rutinas y validación de datos que contribuyen a la elaboración de informes para el análisis de la información requerida para la toma de decisiones.

7 VERIFICACION DEL CUMPLIMIENTO DE LOS OBJETIVOS

Los objetivos planteados en el proyecto se cumplieron en su totalidad durante su desarrollo y son descritos a continuación:

Tabla 4. Cumplimiento de objetivos

Objetivo	Factor de cumplimiento	%
Implementar un sistema de información para la administración y gestión primaria de la micro empresa Dipreco Ltda., que incluya las funciones básicas de inventarios, cuentas por cobrar y nómina.	Especificación de la interfaz (numeral 5.2)	100%
Modelar una solución informática a los problemas de procesamiento y organización de la información en la microempresa Dipreco Ltda, mediante la integración de conceptos de programación y teoría de Bases de Datos, aplicables en un lenguaje Visual y en una herramienta de gestión de base da datos.	Modelo de datos (numeral 5.1)	100%
Desarrollar una herramienta software que facilite el proceso de control de inventarios, el seguimiento de las cuentas por cobrar y el registro de los salarios devengados por cada empleado.	Diseño global, diseño detallado e implementación (numerales 4.6,4.7 y 5)	100%

8 CONCLUSIONES

Se diseñó e implemento un sistema software que permite resolver un problema puntual en manejo de información. Implementa las funciones básicas para el manejo de inventarios, pedidos, actividades en la planta de producción y cuentas por cobrar, basado en los requerimientos descritos por los usuarios del sistema, diagramas de casos de uso, diagramas de actividad, observaciones e innovaciones resultantes de las pruebas según el modelo de prototipado evolutivo.

El software desarrollado ofrece al usuario la posibilidad de realizar consultas y reportes para obtener información actualizada sobre los diferentes procesos llevados en la planta, ya sea clientes morosos o evolución del contrato activo y desempeño de los empleados, crear nuevos usuarios, y mantener actualizados los inventarios.

El sistema permite elaborar reportes de ventas, historial de pedidos y productos pendientes, detalles de contratos, nómina, movimientos en el inventario y cuentas por cobrar, además de generar listas imprimibles de productos, clientes y empleados con sus respectivos detalles. Además permite la impresión de cotizaciones y facturas, gracias al manejo de las herramientas disponibles del lenguaje visual utilizado para el desarrollo y la base de datos elaborada para este sistema.

A nivel personal el crecimiento fue la experiencia de compartir opiniones con personal diferente a nuestra profesión, sentir el apoyo y colaboración en el desarrollo del proyecto de la entidad interesada en mejorar sus actividades, conocer el funcionamiento a fondo de la empresa, tener la responsabilidad de llenar las expectativas iniciales y entregar un producto y una solución conforme a las necesidades que satisfagan y mejoren cada proceso elaborado manualmente en el momento del control de inventarios, seguimiento de cuentas por cobrar y cálculo de nómina.

9 RECOMENDACIONES


Como mejora y complemento del sistema sería adecuado agregarle las opciones de costos y materiales. Para la elaboración de los productos se utilizan varios materiales como cemento, arena, etc. estos costos de producción varían dependiendo de los precios y disponibilidad de estos materiales en el mercado. Se pueden implementar funciones de consulta, inserción y modificación de estos costos, que influyen directamente en los precios de venta, y así facilitar la actualización de estos precios. Otros factores que se pueden tener en cuenta son la variación de los costos de las actividades, por ejemplo en la actividad transporte, el costo depende del precio de la gasolina en ese momento, en este aspecto se podría ampliar la función de consulta de estos costos a un análisis detallado de cada uno de estos.

Como mantenimiento a la base de datos, se requiere que el usuario administrador haga uso adecuado y periódico de la opción de copia de la base de datos.


ANEXO A

DOCUMENTACIÓN DEL MODELO DE DATOS


Tabla Productos contiene información sobre los productos que se producen en la empresa

TABLA PRODUCTOS			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCIÓN
 IdProducto IdCategoria Unidad Nombre Costo PVP PV Contrato Civil PV Contrato Suministro Cantidad en Bodega	Autonumérico Numérico Texto Texto Numérico Numérico Numérico Numérico Numérico	Entero Entero largo Doble Doble Doble Doble Entero largo	Código del producto Nombre del producto


La tabla categorías contiene información de las categorías existentes que clasifican los diferentes productos.

TABLA CATEGORIAS			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCIÓN
 IdCategoria Nombre	Autonumérico Texto	Entero	Código de categoría Nombre de la categoría



La tabla actividades contiene las diferentes actividades que se realizan en la planta.

TABLA ACTIVIDADES			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCIÓN
 IdActividad Tipo	Autonumérico Texto	Entero	Código de actividad Nombre de la actividad





La tabla producto_actividad representa una relación uno a muchos con la tabla productos y la tabla actividades y contiene información del precio que se paga por actividad para cada producto.

TABLA PRODUCTO_ACTIVIDAD			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCIÓN
 IdProducto	Numérico	50	Código de producto
 IdActividad	Numérico	Entero	Código de actividad
Valor Unitario	Numérico	Doble	Valor unitario de la actividad por producto


La tabla movimientos contiene la información de los movimientos diarios en el inventario por producto.

TABLA MOVIMIENTOS			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCIÓN
 IdMov	Autonumérico	Entero	Código del movimiento
 IdProducto	Numérico	Entero	Código del producto
Fecha	Fecha	Date	Fecha del movimiento
Entrada	Numérico	Entero	Cantidad entrada en invent.
Salida	Numérico	Entero	Cantidad salida en invent.
Saldo	Numérico	Entero	Saldo que queda en invent.
Desperdicio	Numérico	Entero	Salida por desperdicio


La tabla elaborado es una relación de uno a muchos con las tablas productos, actividades, contratos y empleados y contiene información sobre las actividades realizadas durante el día.

TABLA ELABORADO			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCIÓN
 IdItem	Autonumérico	Entero largo	Código de elaborado
 IdCont	Numérico	Entero largo	Código de contrato
 IdEmp	Numérico	Entero largo	Código de empleado
 IdPro	Numérico	Entero largo	Código de producto
 IdAct	Numérico	Entero largo	Código de actividad
Fecha	Date	Fecha corta	Fecha de registro
Cantidad	Numérico	Entero	Cantidad de producto
Valor	Numérico	Doble	Valor a pagar al empleado por la actividad

La tabla contratos corresponde a los contratos creados semanalmente.

TABLA CONTRATOS			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCIÓN
 IdContrato	Autonumérico	Entero	Código de contrato
FechaIni	Date	Fecha corta	Fecha inicial del contrato
FechaTermi	Date	Fecha corta	Fecha final del contrato
Valor	Numérico	Doble	Valor del contrato



La tabla empleados contiene información de los empleados de la planta.

TABLA EMPLEADOS			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCIÓN
 IdEmpleado	Autonumérico	Entero largo	Id del empleado
Nombre	Texto		Nombre del empleado
Dirección	Texto		Dirección del empleado
Telefono1	Numérico	Entero	Teléfonos del empleado
Telefono2	Numérico	Entero	
Celular	Numérico	Entero	

La tabla pedidos corresponde a los ítems de una venta y representa una relación uno a muchos con la tabla productos y la tabla ventas.

TABLA PEDIDOS			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCIÓN
 IdItem	Autonumérico	Entero largo	Código de pedido
 IdVenta	Numérico	Entero largo	Código de venta
 IdPro	Numérico	Entero largo	Código de producto
Cantidad	Numérico	Entero largo	Cantidad de producto
Valor Unitario	Numérico	Doble	Precio del producto (PVP, PVCCivil...)



La tabla ventas corresponde al registro de una venta realizada a un determinado cliente, contiene información sobre fecha de pedido y de entrega del mismo, valor total de la venta, iva, descuentos, etc.

TABLA VENTAS			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCIÓN
 IdVenta	Autonumérico	Entero largo	Código de venta
 IdCliente	Numérico	Entero largo	Código del cliente
Fecha	Date	Fecha corta	Fecha del pedido
Fecha Entrega	Date	Fecha corta	Fecha de entrega de pedido
Iva	Numérico	Doble	Iva
Descuento	Numérico	Doble	Descuento
Anticipo	Numérico	Doble	Anticipo dado por el cliente
Valor Venta	Numérico	Doble	Valor total de la venta
Pendiente	Si/No	Activo/Inactivo	Pedido pendiente

La tabla clientes contiene la información de los clientes.

TABLA CLIENTES			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCIÓN
 IdCliente Nombre Dirección Telefono1 Celular	Autonumérico Texto Texto Numérico Numérico	Entero largo Entero Entero	Código del cliente Nombre del cliente Dirección del cliente Teléfonos del cliente

La tabla CxC corresponde a las cuentas por cobrar, esta relacionada con la tabla clientes y contiene la fecha de pago y el valor a pagar.

TABLA C x C			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCIÓN
 IdCuentas  IdCli Fecha Valor	Autonumérico Numérico Date Numérico	Entero largo Entero Fecha corta Doble	Código de la cuenta Código del cliente Fecha de pago de cuenta Valor de la cuenta a pagar

La tabla usuarios contiene información de los usuarios del sistema.

TABLA USUARIOS			
NOMBRE DEL CAMPO	TIPO DE DATO	FORMATO	DESCRIPCION
 IdUsuario NombreUsuario Clave	Autonumérico Texto Texto	Entero	Identificación de usuario Nombre de acceso al sistema del Usuario Clave de acceso al sistema del usuario

BIBLIOGRAFIA

PIATTINI, Mario. Análisis y diseño detallado de aplicaciones informáticas de gestión. México D.F: Alfa y Omega, 2000.

MCCONNELL, Steve. Desarrollo y Gestión de Proyectos Informáticos. España: McGraw Hill, 1996.

PRESSMAN, Roger, Ingeniería del software: Un Enfoque Práctico. Quinta edición. España: McGraw Hill, 2002.

GRECH, Pablo. Introducción a la ingeniería. 1ª Edición. Bogotá: Prentice Hall, 2001.

GOMEZ, Luis Carlos. Planeación de proyectos: Un Enfoque para la Ingeniería de Sistemas e informática.

INSTITUTO COLOMBIANO DE NORMAS TECNICAS Y CERTIFICACION. Documentación. Presentación de tesis, trabajos de grado y otros trabajos de investigación. Bogotá. ICONTEC, 2002. (NTC 1486)

SOM, Guillermo. Curso de iniciación a la programación con Visual Basic .NET. 2001.

Disponible en Internet:

<<http://www.mundoprogramacion.com/NET/cursoVB.NET/indice.htm>>

Manual de VB.Net Avanzado.

Disponible en Internet: <<http://www.programatium.com/vbnet-avanzado.htm>>

MARTINEZ MONTEJO, Enrique. Acceso a bases de datos: ADO.Net. 2003

Disponible en Internet: <<http://mvp-access.com/softjaen>>

MSDN Online en Español: Artículos Visual Basic .Net

Disponible en Internet:

http://www.microsoft.com/spanish/msdn/articulos/architectema/tema/visual_basica.aspx

Ivethe Rocio Monroy Blanco