

INFORME FASE III – PROYECTO FINAL

TITULO: Sistema de Gestión de Facturación Empresarial (SGFE)

José Fernando López
Plan: 3743
Código: 1035204
Correo: j_fernando1010@hotmail.com

Christian Santander
Plan: 3743
Código: 1029925
Correo: christhian0512@gmail.com

Yeferson Guarín
Plan: 3743
Código: 1037285
Correo: Jeff.aristi@gmail.com

Sebastián Vargas
Plan: 3743
Código: 1034057
Correo: sevarmo@gmail.com

Base de Datos I
Universidad del Valle
Junio 2013

RESUMEN

El Almacén de Variedades “Nataly” desea automatizar su proceso interno de pagos y ventas: este proyecto se centra en investigar los aspectos que enmarca este proceso para poder obtener un documento estructurado y una aplicación que permita generar ventas, realizar pagos a proveedores y pagos de clientes, así como gestionar la información de los proveedores, clientes, productos y empleados del almacén para luego guardarlas en forma ordenada para una posterior consulta, cambio o eliminación. Se pretende garantizar el cumplimiento y entrega frente a los requerimientos del usuario.

En este proyecto se encuentran varios aspectos que toman como eje principal la ingeniería de software. El entorno de desarrollo en que se trabajó el proyecto fue JAVA como lenguaje de programación, y usando una base de datos creada y administrada con el Sistema de Gestión de Bases de Datos PostgreSQL.

Este software funciona como una ayuda que permite automatizar y mejorar el proceso de ventas, la ejecución de pagos y adquisición de productos a proveedores, la administración de productos, clientes, empleados y proveedores.

El proyecto se centrará en atender una necesidad primordial dentro de una organización, que tendrá como eje principal el diseño, desarrollo y adaptación de un sistema de información que permita manejar varios datos y optimizar el proceso de compras y ventas.

Conscientes del problema que presentan muchos Almacenes en cuanto a la optimización, control y administración de sus procesos, se ha hecho énfasis especialmente en una empresa dedicada a diversos aspectos como lo son los servicios de compra y venta de artículos.

INTRODUCCIÓN

Por medio de este proyecto se desea analizar, estudiar y comprender el proceso de facturación del Almacén de Variedades “Nataly”. En este proyecto se pretende hacer una fuente de investigación para obtener los requerimientos básicos del sistema (aspectos funcionales y no funcionales). En este proyecto se pretende hacer un análisis de sistema tomando los puntos conceptuales del usuario para así obtener un buen diseño teniendo en cuenta los conceptos vistos en el curso de Base de Datos.

Este proyecto comprende todos los aspectos fundamentales que marcó la investigación que consta de: un ámbito general de trabajo. Se hace una descripción del problema, las causas y los objetivos. De esta forma estudiaremos el problema desde un punto de vista técnico y organizacional.

En la segunda parte se conocerán los aspectos más elementales del Almacén, como son: su funcionamiento, su organización, su historia y sus necesidades de informatización. De esta forma se sabrá cómo funciona el Almacén y se mencionarán los aspectos a trabajar en este proyecto.

Luego se aplicará la ingeniería al proyecto, centrada en el modelo de Desarrollo Ciclo de Vida Clásico en donde se hace un análisis y diseño del proyecto (diagrama Entidad-Relación, esquema Relacional, definición de las dependencias funcionales, definición de las llaves primarias, etc...), donde se pretende obtener un informe teórico y técnico del sistema que se encontró y el sistema a desarrollar. Además se conocerá la estructura del programa, así como las pruebas aplicadas y los aspectos complementarios del proyecto.

TEMA DEL PROYECTO

Para crear un sistema de información que permita automatizar el proceso de compras y ventas de productos en el Almacén, hay que tener en cuenta que son muchos los datos que se van a manejar, teniendo en cuenta que una base de datos relacional es un modelo lógico que se encargan de manipular muchos datos y permite almacenarlos en diferentes tablas relacionadas entre sí para evitar la redundancia y poder compartir los datos.

Para llevar a cabo este proceso se necesita implementar un sistema de Base de Datos Relacional, ya que permite procesar cierto volumen de datos, manejando una simplicidad y sencillez en el procesamiento de datos. Además la manipulación de los registros y la consulta de cada registro debe ser lo más sencilla posible.

El proyecto se centra en administrar el proceso de compras y ventas de productos mediante una base de datos, permitiendo la inserción, consulta, modificación y eliminación de datos.

Por este motivo, el tema del proyecto se centra en la construcción y adaptación de un sistema de información utilizando un lenguaje de programación.

El Sistema de Gestión de Facturación Empresarial (SGFE), se desarrollará para el Almacén Variedades "Nataly".

FORMULACIÓN DEL PROBLEMA

El Almacén “Variedades Nathaly” trabaja especialmente en la venta de prendas de vestir para damas y caballeros. La necesidad de procesar y manipular la información de todas las ventas, compras y en sí la información general de productos, clientes, proveedores y empleados de una manera más sencilla y precisa es alta. Lo que realmente desean alcanzar es una optimización en los procesos de compras y ventas de productos.

Un inconveniente encontrado fue la dificultad que tienen al generar una sencilla venta de un producto, ya que no tienen un formato digital estándar y simplemente se basan en un formato físico. Por este motivo, es muy importante distribuir, organizar y administrar la información que involucren estos entes, para que así alcanzar una mayor efectividad al generar las facturas y simplificar del problema, dando mejor organización y mayor orden al manejo de dicha información.

OBJETIVOS

Objetivo General:

Desarrollar un sistema de información que permita automatizar el proceso de compras y ventas y administrar la información de los clientes, productos, empleados y proveedores que tiene a su disposición el Almacén de Variedades “Nataly”

Objetivos Específicos:

- Estudiar la información existente en el Almacén, de tal forma que se puedan detectar las fallas y soluciones que ameriten al desarrollo de un sistema de información.
- Gestionar la información de los Clientes, Productos, Empleados, Proveedores y Compras y Ventas de Artículos.
- Manejar el proceso de registro, y consulta de clientes, empleados, proveedores y productos.
- Manejar el proceso de ventas y compras.
- Manejar el proceso de pagos de clientes y a proveedores.

ANTECEDENTES DEL ALMACÉN

Funcionamiento: Variedades Nataly está legalmente constituida, se encuentra ubicada en el centro de Cali, centro comercial el Caleño. Tiene varios locales, cada uno, se enfoca en una línea. Atiende de lunes a sábado de 8:00AM a 7:00PM. Todos los registros son manejados manualmente.

Organización: Es dirigida por un administrador, quién es el dueño. En cada local hay un vendedor principal y uno o dos vendedores de apoyo. Hay un bodeguero que maneja la bodega

Fundamentos del proyecto: La empresa lleva ya varios años en el mercado, y ha tenido éxito, pero se ha visto que el manejo de archivos es cada vez más complejo y muy susceptible a errores, además son cada vez más las herramientas que dan solución a este problema.

ASPECTOS METODOLÓGICOS

Tiempo: El tiempo estipulado para la realización del proyecto fue de 2 meses y medio (75 días)

Hardware: Para el desarrollo, adaptación y prueba del software, es necesario que el equipo disponga de ciertas especificaciones de hardware.

Procesador: Pentium III - 3.1.4 GHZ

Memoria RAM: 512MB

Disco Duro: 180GB

Resolución de Pantalla: 1024x768

Software: Para realizar el proyecto se requirió del IDE Netbeans, Sistema de Gestión de Bases de Datos PostgreSQL y el administrador de base de datos PG Admin.

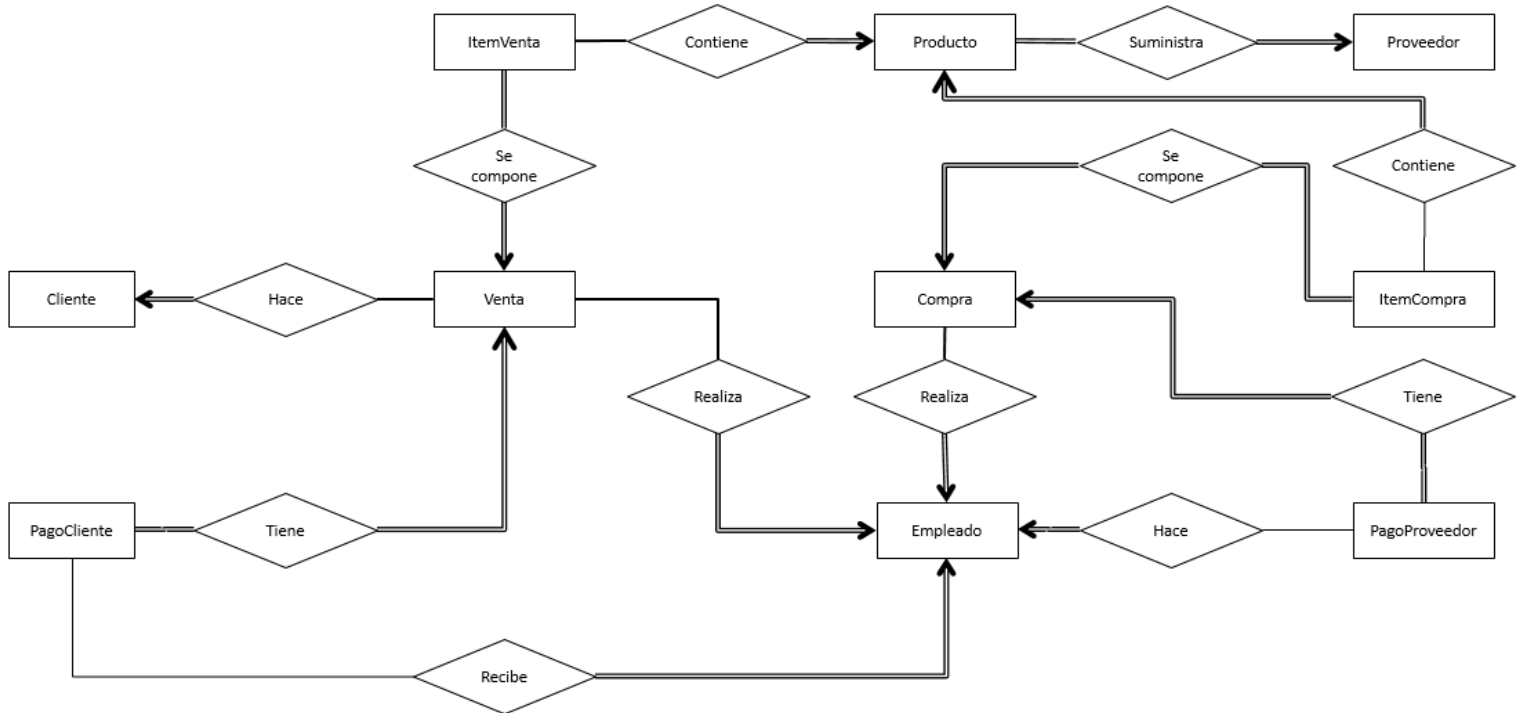
Costos: Dado que el programa fue realizado por estudiantes de la universidad y el proyecto se realizó en software libre no se presentaron costos.

Personal: Los integrantes del grupo estábamos capacitados para la realización de este proyecto al integrar conocimientos de base de datos, desarrollo de software, diseño de interfaces, modelo vista controlador, todo esto visto durante el transcurso de la carrera.

Lugar: Se trabajó tanto en la universidad como en las respectivas casas de los integrantes, esto no favoreció ni perjudicó al desarrollo correcto del proyecto.

Diseño de la Aplicación.

- Diagrama Entidad-Relación definitivo de la base de datos.



- Esquema relacional de la base de datos de acuerdo con su Diagrama E/R
- Cliente(IdCilente,Nombre,Apellido,Direccion,Telefono,saldo)
 - Empleado(IdEmpleado,Nombre,Apellido,Direccion,Telefono,cargo)
 - Proveedor(IdProveedor,Nombre,Direccion,Telefono,saldo)
 - Producto(IdProducto,Descripcion,IdProveedor,precioCompra,precioVenta,cantDisponible)
- Llave foránea: IdProveedor referencia IdProveedor en Proveedor
- Venta(idVenta,idEmpleado,idCliente,fechaVenta, vlrTotal,saldoVenta)
- Llave foránea: idEmpleado referencia idEmpleado en Empleado
- Llave foránea: idCliente referencia idCliente en Cliente
- ItemVenta(nroItem,idVenta,idProducto,Cantidad,iva,vlrTotal)
- Llave foránea: idProducto referencia idProducto en Producto
- Llave foránea: idVenta referencia idVenta en Venta
- PagoCliente(nroPago,idVenta,fecha,monto,idEmpleado)
- Llave foránea: idVenta referencia idVenta en Venta
- Llave foránea: idEmpleado referencia idEmpleado en Empleado

-Compra(idCompra,idProveedor,idEmpleado,fechaCompra,vlrTotal,saldoCompra)

Llave foránea: IdProveedor referencia IdProveedor en Proveedor

Llave foránea: idEmpleado referencia idEmpleado en Empleado

Llave foránea: idProducto referencia idProducto en Producto

-ItemCompra(nroItem,idCompra,idProducto,Cantidad,vlrTotal)

Llave foránea: idCompra referencia idCompra en Compra

Llave foránea: idEmpleado referencia idEmpleado en Empleado

Llave foránea: idProducto referencia idProducto en Producto

-PagoProveedor(nroPago,idCompra,fecha,monto,idEmpleado)

Llave foránea: idCompra referencia idCompra en Compra

Llave foránea: idEmpleado referencia idEmpleado en Empleado

- Especificación de las llaves foráneas

Llave foránea: IdProveedor en Producto referencia IdProveedor en Proveedor

Llave foránea: idEmpleado en Venta referencia idEmpleado en Empleado

Llave foránea: idCliente en Venta referencia idCliente en Cliente

Llave foránea: idProducto en ItemVenta referencia idProducto en Producto

Llave foránea: idVenta en ItemVenta referencia idVenta en Venta

Llave foránea: idVenta en PagoCliente referencia idVenta en Venta

Llave foránea: idEmpleado en PagoClientereferencia idEmpleado en Empleado

Llave foránea: IdProveedor en Compra referencia IdProveedor en Proveedor

Llave foránea: idEmpleado en Compra referencia idEmpleado en Empleado

Llave foránea: idProducto en Compra referencia idProducto en Producto

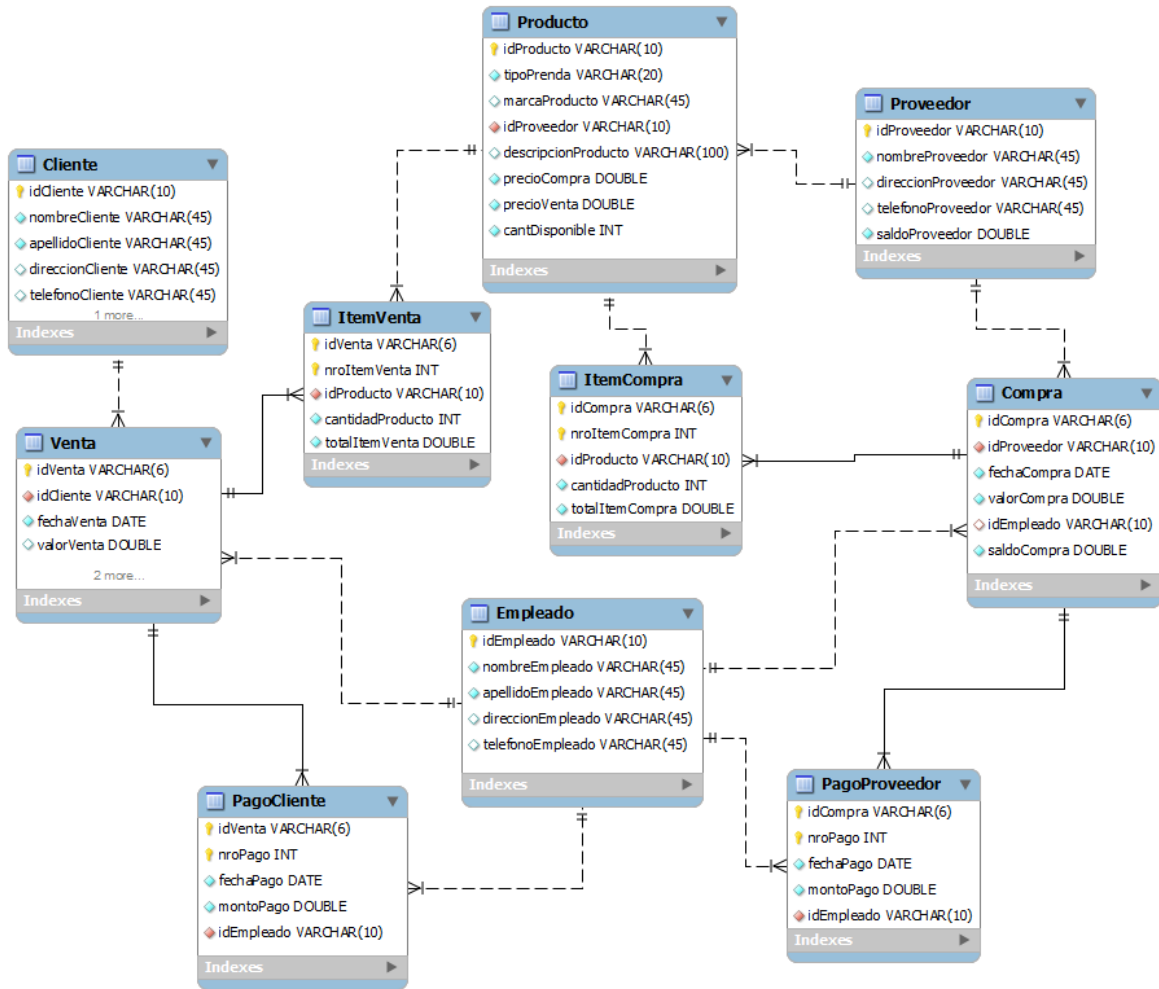
Llave foránea: idCompra en ItemCompra referencia idCompra en Compra

Llave foránea: idEmpleado en ItemCompra referencia idEmpleado en Empleado

Llave foránea: idProducto en ItemCompra referencia idProducto en Producto

Llave foránea: idCompra en PagoProveedor referencia idCompra en Compra

Llave foránea: idEmpleado en PagoProveedor referencia idEmpleado en Empleado



DEPENDENCIAS FUNCIONALES

Venta

IdVenta -> IdEmpleado, IdCliente, FechaVenta, VlrTotal, SaldoVenta

IdEmpleado, IdCliente, FechaVenta -> VlrTotal, SaldoVenta

Proveedor

IdProveedor -> NombreProveedor, DireccionProveedor, TelefonoProveedor, SaldoProveedor

ItemCompra

NroItem, IdCompra -> IdProduct, cantidad, VlrTotal

Empleado

IdEmpleado -> NombreEmpleado, ApellidoEmpleado, DireccionEmpleado, TelefonoEmpleado, CargoEmpleado

Cliente

IdCliente -> NombreCliente, ApellidoCliente, DireccionCliente, TelefonoCliente, SaldoCliente

Compra

IdCompra -> IdProveedor, IdEmpleado, FechaCompra, VlrTotal, SaldoCompra
IdProveedor, IdEmpleado -> FechaCompra, VlrTotal, SaldoCompra

Producto

Idproducto -> Descripcion, PrecioCompra, PrecioVenta, CantDisponible, IdProveedor

PagoProveedor

NroPago, IdCompra-> FechaPago, Monto, IdEmpleado

PagoCliente

NroPago, IdVenta -> FechaPago, Monto, IdEmpleado

ItemVenta

NroItem, IdVenta -> IdProducto, Cantidad, Iva, VlrTotal

Claves**Proveedor**

Paso 1 se eliminan todos los atributos independientes
no hay atributos independientes asi que no se eliminan y R = Rsi

Paso 2 se escoge uno y se eliminan descriptores equivalentes
no se encuentran descriptores equivalentes asi que no se eliminan y Rsi = Rsie

Paso 3 descriptor en el que no haya implicados
en este caso IdProveedor no es implicado así que puede formar parte de la clave
 $K_p = \text{IdProveedor}$

Paso 4 descriptor clave implicado
en el cierre de $k_p \rightarrow k_{p+}$ están todos los atributos y no se usan en una dependencia diferente
a las usadas para calcular K_{p+}

Clave IdProveedor

ItemCompra

Paso 1 se eliminan todos los atributos independientes
no hay atributos independientes así que no se eliminan y $R = R_{si}$

Paso 2 se escoge uno y se eliminan descriptores equivalentes
no se encuentran descriptores equivalentes así que no se eliminan y $R_{si} = R_{sie}$

Paso 3 descriptor en el que no haya implicados
en este caso NroItem y IdCompra no están implicados así que pueden formar parte de las
claves
 $K_p = \text{NroItem, IdCompra}$

Paso 4 descriptor clave implicado
en el cierre de $k_p \rightarrow k_{p+}$ están todos los atributos y no se usan en una dependencia diferente
a las usadas para calcular K_{p+}

Clave NroItem, IdCompra

Empleado

Paso 1 se eliminan todos los atributos independientes
no hay atributos independientes así que no se eliminan y $R = R_{si}$

Paso 2 se escoge uno y se eliminan descriptores equivalentes

no se encuentran descriptores equivalentes así que no se eliminan y $R_{si} = R_{sie}$

Paso 3 descriptor en el que no haya implicados

en este caso $IdEmpleado$ no está implicado así que puede formar parte de la clave

$K_p = IdEmpleado$

Paso 4 descriptor clave implicado

en el cierre de K_p K_p^+ están todos los atributos y no se usan en una dependencia diferente a las usadas para calcular K_p^+

Clave $IdEmpleado$

Cliente

Paso 1 se eliminan todos los atributos independientes

no hay atributos independientes así que no se eliminan y $R = R_{si}$

Paso 2 se escoge uno y se eliminan descriptores equivalentes

no se encuentran descriptores equivalentes así que no se eliminan y $R_{si} = R_{sie}$

Paso 3 descriptor en el que no haya implicados

en este caso $IdCliente$ no está implicado así que puede formar parte de la clave

$K_p = IdCliente$

Paso 4 descriptor clave implicado

en el cierre de K_p K_p^+ están todos los atributos y no se usan en una dependencia diferente a las usadas para calcular K_p^+

Clave $IdCliente$

Producto

Paso 1 se eliminan todos los atributos independientes

no hay atributos independientes así que no se eliminan y $R = R_{si}$

Paso 2 se escoge uno y se eliminan descriptores equivalentes
no se encuentran descriptores equivalentes así que no se eliminan y $R_{si} = R_{sie}$

Paso 3 descriptor en el que no haya implicados
en este caso $Id_{producto}$ no está implicado así que puede formar parte de la clave
 $K_p = Id_{producto}$

Paso 4 descriptor clave implicado
en el cierre de K_p K_{p+} están todos los atributos y no se usan en una dependencia diferente
a las usadas para calcular K_{p+}

Clave $Id_{producto}$

PagoProveedor

Paso 1 se eliminan todos los atributos independientes
no hay atributos independientes así que no se eliminan y $R = R_{si}$

Paso 2 se escoge uno y se eliminan descriptores equivalentes
no se encuentran descriptores equivalentes así que no se eliminan y $R_{si} = R_{sie}$

Paso 3 descriptor en el que no haya implicados
en este caso Nro_{Pago} e Id_{Compra} no están implicados así que pueden formar parte de las
claves
 $K_p = Nro_{Pago}, Id_{Compra}$

Paso 4 descriptor clave implicado
en el cierre de K_p K_{p+} están todos los atributos y no se usan en una dependencia diferente
a las usadas para calcular K_{p+}

Clave Nro_{Pago}, Id_{Compra}

PagoCliente

Paso 1 se eliminan todos los atributos independientes
no hay atributos independientes así que no se eliminan y $R = R_{si}$

Paso 2 se escoge uno y se eliminan descriptores equivalentes
no se encuentran descriptores equivalentes así que no se eliminan y $R_{si} = R_{sie}$

Paso 3 descriptor en el que no haya implicados
en este caso NroPago e IdVenta no están implicados así que pueden formar parte de las claves
 $K_p = \text{NroPago}, \text{IdVenta}$

Paso 4 descriptor clave implicado
en el cierre de k_p k_{p+} están todos los atributos y no se usan en una dependencia diferente a las usadas para calcular K_{p+}

Clave NroPago, IdVenta

Itemventa

Paso 1 se eliminan todos los atributos independientes
no hay atributos independientes así que no se eliminan y $R = R_{si}$

Paso 2 se escoge uno y se eliminan descriptores equivalentes
no se encuentran descriptores equivalentes así que no se eliminan y $R_{si} = R_{sie}$

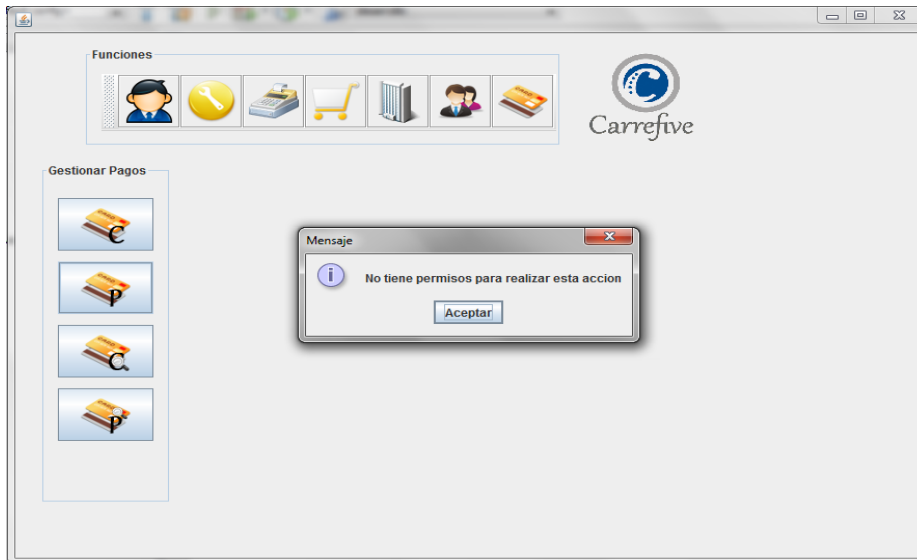
Paso 3 descriptor en el que no haya implicados
en este caso NroItem e IdVenta no están implicados así que pueden formar parte de las claves
 $K_p = \text{NroItem}, \text{IdVenta}$

Paso 4 descriptor clave implicado
en el cierre de k_p k_{p+} están todos los atributos y no se usan en una dependencia diferente a las usadas para calcular K_{p+}

Clave NroItem, IdVenta

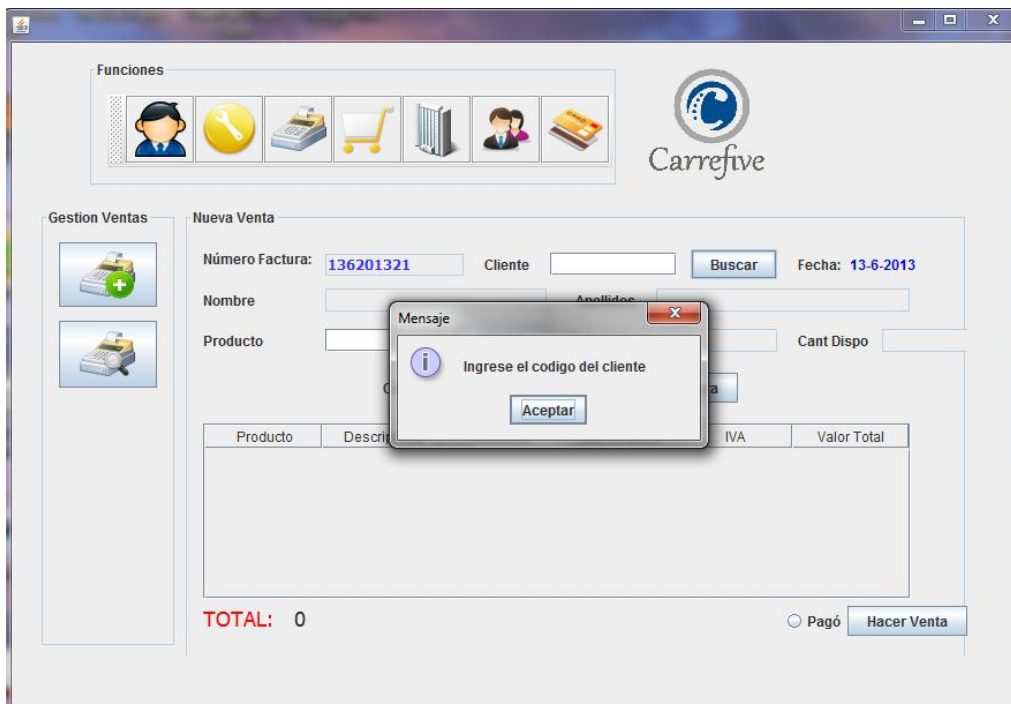
- Pruebas de la aplicación (valores ingresados, resultados esperados y resultados obtenidos)

Permisos

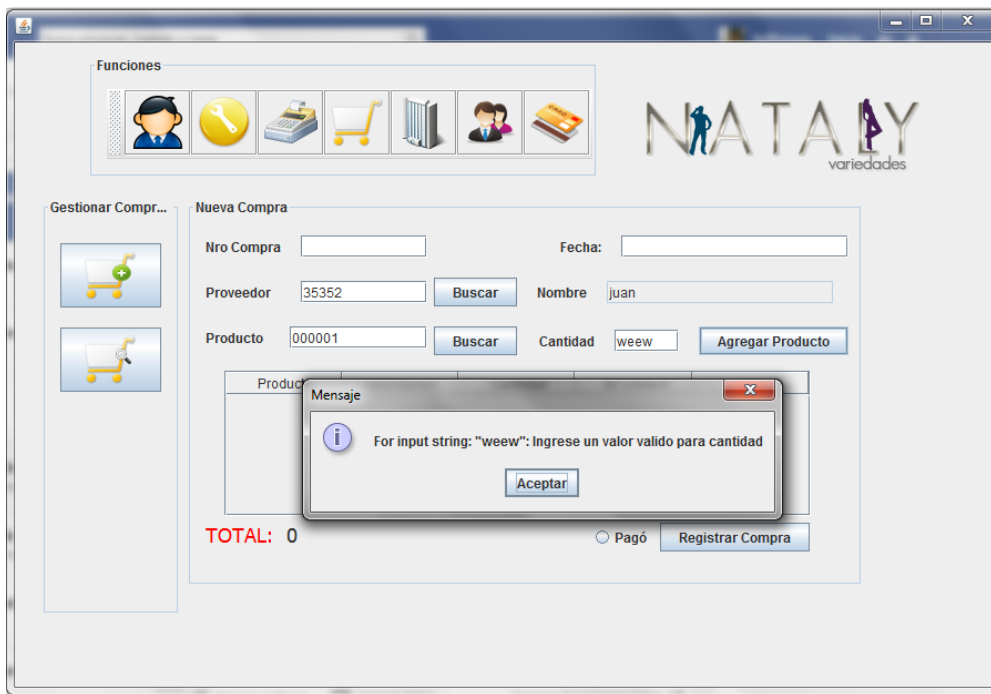


Validacion

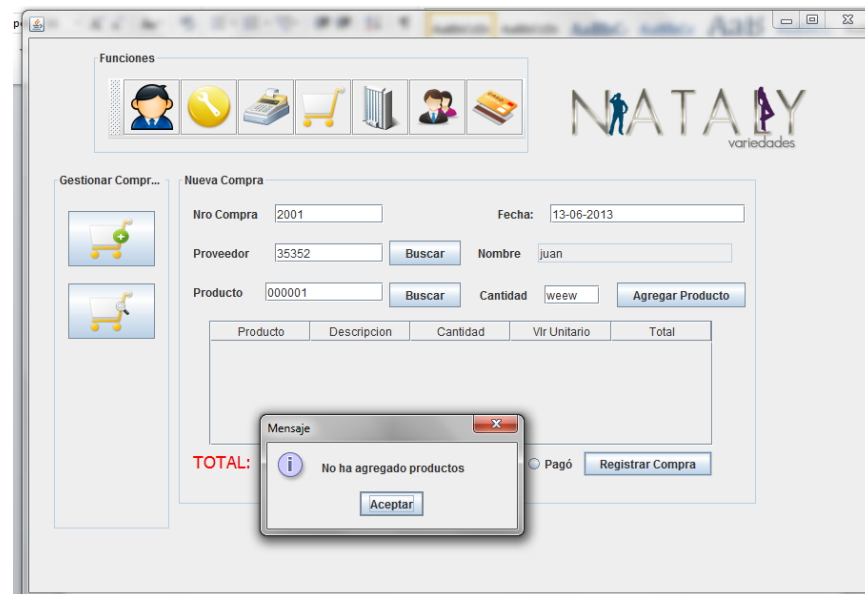
-Campos vacios



-Valores incorrectos



-Funcionamiento correcto



Funcionalidad de uno de los módulos

The screenshot shows the 'Nueva Venta' (New Sale) module in the Carrefive system. The interface includes a 'Funciones' (Functions) bar with icons for user management, settings, sales, inventory, reports, and other functions. The 'Gestion Ventas' (Manage Sales) section on the left has icons for adding new sales and viewing existing ones. The main form for 'Nueva Venta' contains the following fields:

- Número Factura:** 136201321
- Cliente:** 1144163369
- Fecha:** 13-6-2013
- Nombre:** jeferson
- Apellidos:** guarin
- Producto:** 000001
- Precio Venta:** 20000.0
- Cant Dispo:** 2
- Cantidad:** 1

A 'Buscar' (Search) button is located next to the 'Producto' field. An 'Agregar a Factura' (Add to Invoice) button is positioned below the 'Cantidad' field. A table displays the current sale item:

Producto	Descripción	Cantidad	Valor Unitario	IVA	Valor Total
000001	pantalon	1	20000.0	0.0	20000.0

The total amount is displayed as **TOTAL: 20000.0**. At the bottom right, there are radio buttons for 'Pagó' (Paid) and a 'Hacer Venta' (Make Sale) button.

This screenshot shows the same 'Nueva Venta' module interface as the previous one, but with a success message overlay. The message box, titled 'Mensaje', contains the text: 'La Venta136201351 se registro con exito' (The Sale 136201351 was registered successfully). An 'Aceptar' (Accept) button is located at the bottom of the message box. The underlying form and table are identical to the previous screenshot.

Carrefive

Gestion Ventas

Buscar

136201351

Buscar

Consultar Venta

Fecha Venta

2013-06-13

Nro Venta

136201351

Cliente

1144163369

Empleado

2344234

Total

20000.0

Saldo

20000.0

Producto	Descripción	Cantidad	Valor Unita.	IVA	Valor Total
000001	pantalon	1	20000.0	0.0	20000.0

ANÁLISIS

Para poder automatizar el proceso de facturación y compras de productos en el Almacén Variedades “Nathaly” se usará una base de datos que se encargue de almacenar y administrar la información a través de la aplicación en el lenguaje JAVA desarrollada. De esta forma, los datos de los clientes, productos, proveedores, empleados, ventas y compras se manejarán a través de un Sistema de Gestión de Bases de Datos.

Gracias a esta aplicación se podrá agilizar la generación de pagos y ventas, la administración de clientes, productos, proveedores y empleados. Esta aplicación se desarrollará trabajando con la base de datos y se adaptará mediante un entorno facilitando el acceso al usuario y la interacción con el sistema.

Este sistema permite administrar las generalidades de una base de datos (ingresar datos, consultar datos, modificar y listar datos) y permite administrar los registros de los clientes, productos, proveedores, empleados, ventas y compras.

Las consultas y el acceso a la base de datos del sistema se harán mediante un entorno orientado a objetos y el manejo de una Interfaz Gráfica de Usuario (GUI) debe ser amigable y flexible para el usuario, permitiendo una mayor interacción con la aplicación. La aplicación permitirá manipular los datos antiguos y modificarlos siempre y cuando el usuario lo vea necesario.

El **Sistema de Gestión de Facturación Empresarial** está constituido por una base de datos que se encarga de manejar la información de todos los clientes, productos, proveedores, empleados y del proceso de facturación (pagos y ventas). Este sistema se compone de 7 módulos y son los siguientes:

- **Módulo de Clientes.**
- **Módulo de Productos.**
- **Módulo de Ventas.**
- **Módulo de Compras.**
- **Módulo de Proveedores.**
- **Módulo de Empleados.**
- **Módulo de Pagos.**

CONCLUSIONES

Gracias al desarrollo y adaptación de la aplicación, se logró automatizar el proceso de ventas. Teniendo en cuenta que el almacén estudiado utiliza un proceso manual, esta aplicación agilizará el manejo de la información y la generación de ventas, permitiéndole al usuario administrar sus datos de una forma más eficiente, ágil y segura.

Gracias a este proyecto, se pudo detectar que hay muchas pequeñas empresas que quieren ser más eficientes y más competitivas en el mercado. El interés en adquirir estas tecnologías sin costo alguno es muy alto. Con este trabajo se puede deducir que el desarrollo de software hacia esas empresas necesita un mayor impulso y mayor compromiso de las empresas, para poder así construir aplicaciones diseñadas para atender las prioridades más importantes de todo tipo de empresa.

La aplicación desarrollada permitió al usuario ingresar, consultar y listar datos de clientes, productos, empleados y proveedores, así como consultar y realizar las ventas y compras. Además su interfaz permitió al usuario interactuar con una aplicación dinámica que se encargó de automatizar el proceso de compras y ventas de productos en el almacén.