

**"Desarrollo de un Sistema de Información para la Gestión de Ventas de producto final para la compañía Kaip'e Alimentos, El Tigre, Estado Anzoátegui"**

Proyecto Socio Tecnológico como requisito parcial para optar al título de Técnico Superior Universitario en Informática

**Docente Tutor:**

Hernández Laleska

**Equipo Investigador:**

Nicolás Alfaro CI: 31.453.119

Alfonzo Trébol CI: 30.933.291

Emilio Pinto CI: 31.210.836

Fredys Navarro CI: 27.928.000

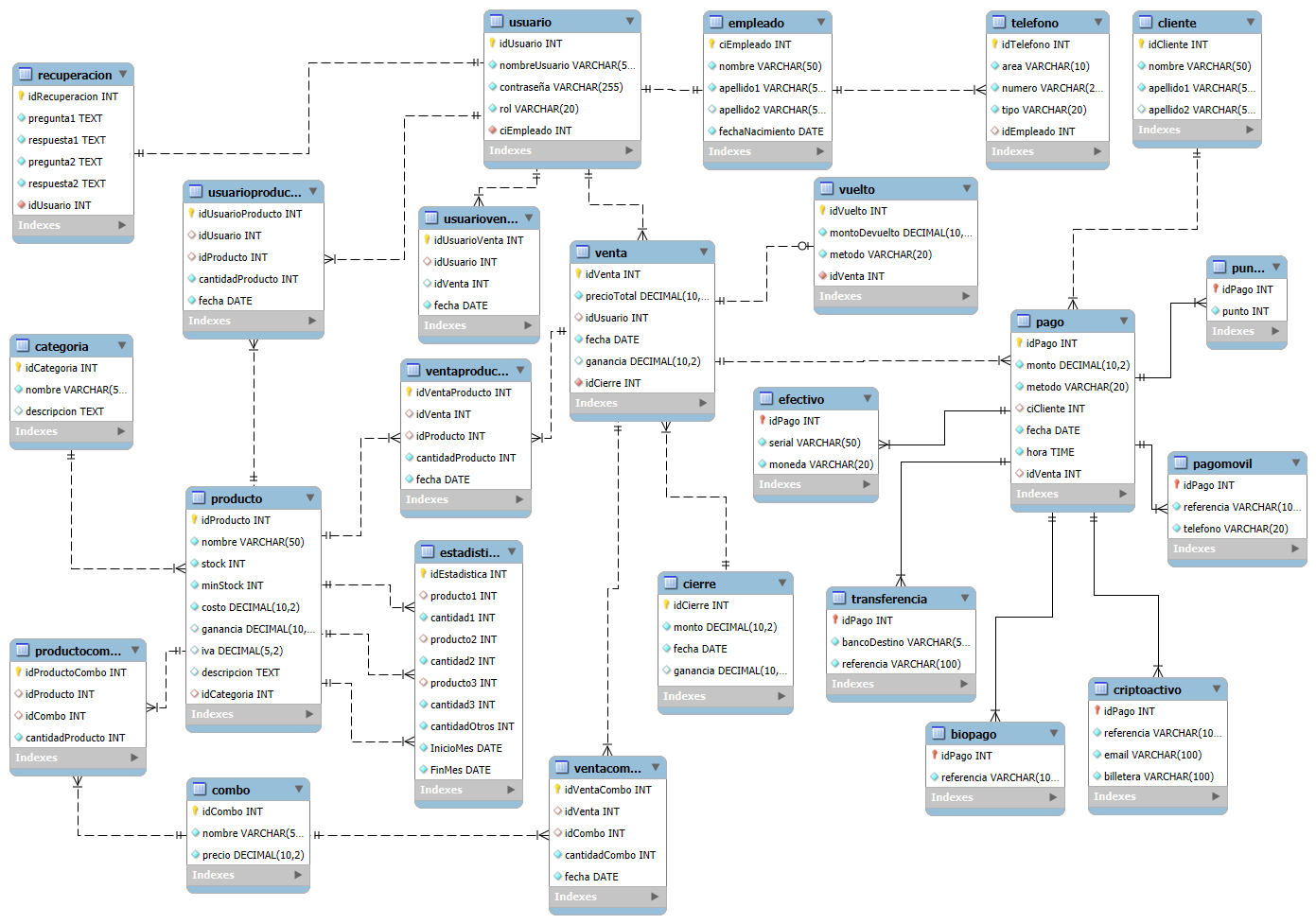
Carlos Navarro CI: 31.468.365

Gabriel Velásquez C.I 31.284.623

El Tigre , julio de 2024

## Diagrama Entidad - Relación (DER)

## Modelo relacional (MR)



***Esquema Relacional***

Empleado (ciEmpleado, nombre, apellido1, apellido2, fechaNacimiento)

Cliente (ciCliente, nombre, apellido1, apellido2)

Categoria (idCategoria, nombre, descripcion)

Producto (idProducto, nombre, stock, minStock, costo, ganancia, iva, descripcion, idCategoria)

Usuario (idUsuario, nombreUsuario, contraseña, rol, ciEmpleado)

Cierre (idCierre, monto, fecha, ganancia)

Venta (idVenta, precioTotal, idUsuario, fecha, ganancia, idCierre)

UsuarioVenta (idUsuarioVenta, idUsuario, idVenta, fecha)

Pago (idPago, monto, metodo, ciCliente, fecha, hora, idVenta)

Recuperacion (idRecuperacion, pregunta1, respuesta1, pregunta2, respuesta2, idUsuario)

Telefono (idTelefono, area, numero, tipo, idEmpleado)

UsuarioProducto (idUsuarioProducto, idUsuario, idProducto, cantidadProducto, fecha)

VentaProducto (idVentaProducto, idVenta, idProducto, cantidadProducto, fecha)

Vuelto (idVuelto, montoDevuelto, metodo, idVenta)

Combo (idCombo, nombre, precio)

VentaCombo (idVentaCombo, idVenta, idCombo, cantidadCombo, fecha)

ProductoCombo (idProductoCombo, idProducto, idCombo, cantidadProducto)

Efectivo (idPago, serial, moneda)

CriptoActivo (idPago, referencia, email, billetera)

PagoMovil (idPago, referencia, telefono)

Transferencia (idPago, bancoDestino, referencia)

BioPago (idPago, referencia)

Punto (idPago, punto)

Estadistica (idEstadistica, producto1, cantidad1, producto2, cantidad2, producto3, cantidad3, cantidadOtros, InicioMes, FinMes)

## .SQL

CREATE DATABASE IF NOT EXISTS `KariñaSystem`;

USE KariñaSystem;

CREATE TABLE IF NOT EXISTS Empleado (

ciEmpleado INT PRIMARY KEY,

nombre VARCHAR(50) NOT NULL,

apellido1 VARCHAR(50) NOT NULL,

apellido2 VARCHAR(50),

fechaNacimiento DATE NOT NULL

);

CREATE TABLE IF NOT EXISTS Cliente (

ciCliente INT PRIMARY KEY,

nombre VARCHAR(50) NOT NULL,

apellido1 VARCHAR(50) NOT NULL,

apellido2 VARCHAR(50)

);

CREATE TABLE IF NOT EXISTS Categoria (

idCategoria INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(50) NOT NULL,

descripcion TEXT

);

CREATE TABLE IF NOT EXISTS Producto (

idProducto INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(50) NOT NULL,

stock INT NOT NULL,

minStock INT NOT NULL,

costo DECIMAL(10, 2) NOT NULL,

ganancia DECIMAL(10, 2),

iva DECIMAL(5, 2),

descripcion TEXT,

idCategoria INT,

FOREIGN KEY (idCategoria) REFERENCES Categoria(idCategoria)

);

CREATE TABLE IF NOT EXISTS Usuario (

idUsuario INT AUTO\_INCREMENT PRIMARY KEY,

nombreUsuario VARCHAR(50) NOT NULL,

contraseña VARCHAR(255) NOT NULL,

rol VARCHAR(20) NOT NULL,

ciEmpleado INT,

FOREIGN KEY (ciEmpleado) REFERENCES Empleado(ciEmpleado)

);

CREATE TABLE IF NOT EXISTS Cierre (

idCierre INT AUTO\_INCREMENT PRIMARY KEY,

monto DECIMAL(10, 2) NOT NULL,

fecha DATE NOT NULL,

ganancia DECIMAL(10, 2)

);

CREATE TABLE IF NOT EXISTS Venta (

idVenta INT AUTO\_INCREMENT PRIMARY KEY,

precioTotal DECIMAL(10, 2) NOT NULL,

idUsuario INT,

fecha DATE NOT NULL,

ganancia DECIMAL(10, 2),

idCierre INT,

FOREIGN KEY (idCierre) REFERENCES Cierre(idCierre),

FOREIGN KEY (idUsuario) REFERENCES Usuario(idUsuario)

);

CREATE TABLE IF NOT EXISTS UsuarioVenta (

idUsuarioVenta INT AUTO\_INCREMENT PRIMARY KEY,

idUsuario INT,

idVenta INT,

fecha DATE NOT NULL,

FOREIGN KEY (idUsuario) REFERENCES Usuario(idUsuario)

);

CREATE TABLE IF NOT EXISTS Pago (

idPago INT AUTO\_INCREMENT PRIMARY KEY,

monto DECIMAL(10, 2) NOT NULL,

metodo VARCHAR(20) NOT NULL,

ciCliente INT,

fecha DATE NOT NULL,

hora TIME NOT NULL,

idVenta INT,

FOREIGN KEY (ciCliente) REFERENCES Cliente(ciCliente),

FOREIGN KEY (idVenta) REFERENCES Venta(idVenta)

);

CREATE TABLE IF NOT EXISTS Recuperacion (

idRecuperacion INT AUTO\_INCREMENT PRIMARY KEY,

pregunta1 TEXT NOT NULL,

respuesta1 TEXT NOT NULL,

pregunta2 TEXT NOT NULL,

respuesta2 TEXT NOT NULL,

idUsuario INT,

FOREIGN KEY (idUsuario) REFERENCES Usuario(idUsuario)

);

CREATE TABLE IF NOT EXISTS Telefono (

idTelefono INT AUTO\_INCREMENT PRIMARY KEY,

area VARCHAR(10) NOT NULL,

numero VARCHAR(20) NOT NULL,

tipo VARCHAR(20) NOT NULL,

idEmpleado INT,

FOREIGN KEY (idEmpleado) REFERENCES Empleado(ciEmpleado)

);

CREATE TABLE IF NOT EXISTS UsuarioProducto (

idUsuarioProducto INT AUTO\_INCREMENT PRIMARY KEY,

idUsuario INT,

idProducto INT,

cantidadProducto INT NOT NULL,

fecha DATE NOT NULL,

FOREIGN KEY (idUsuario) REFERENCES Usuario(idUsuario),

FOREIGN KEY (idProducto) REFERENCES Producto(idProducto)

);

CREATE TABLE IF NOT EXISTS VentaProducto (

idVentaProducto INT AUTO\_INCREMENT PRIMARY KEY,

idVenta INT,

idProducto INT,

cantidadProducto INT NOT NULL,

fecha DATE NOT NULL,

FOREIGN KEY (idVenta) REFERENCES Venta(idVenta),

FOREIGN KEY (idProducto) REFERENCES Producto(idProducto)

);

CREATE TABLE IF NOT EXISTS Vuelto (

idVuelto INT AUTO\_INCREMENT PRIMARY KEY,

montoDevuelto DECIMAL(10, 2) NOT NULL,

metodo VARCHAR(20) NOT NULL,

idVenta INT,

FOREIGN KEY (idVenta) REFERENCES Venta(idVenta)

);

CREATE TABLE IF NOT EXISTS Combo (

idCombo INT AUTO\_INCREMENT PRIMARY KEY,

nombre VARCHAR(50) NOT NULL,

precio DECIMAL(10, 2) NOT NULL

);

CREATE TABLE IF NOT EXISTS VentaCombo (

idVentaCombo INT AUTO\_INCREMENT PRIMARY KEY,

idVenta INT,

idCombo INT,

cantidadCombo INT NOT NULL,

fecha DATE NOT NULL,

FOREIGN KEY (idVenta) REFERENCES Venta(idVenta),

FOREIGN kEY (idCombo) REFERENCES Combo(idCombo)

);

CREATE TABLE IF NOT EXISTS ProductoCombo (

idProductoCombo INT AUTO\_INCREMENT PRIMARY KEY,

idProducto INT,

idCombo INT,

cantidadProducto INT NOT NULL,

FOREIGN KEY (idProducto) REFERENCES Producto(idProducto),

FOREIGN KEY (idCombo) REFERENCES Combo(idCombo)

);

CREATE TABLE IF NOT EXISTS Efectivo (

idPago INT,

serial VARCHAR(50) NOT NULL,

moneda VARCHAR(20) NOT NULL,

PRIMARY KEY (idPago),

FOREIGN KEY (idPago) REFERENCES Pago(idPago)

);

CREATE TABLE IF NOT EXISTS CriptoActivo (

idPago INT,

referencia VARCHAR(100) NOT NULL,

email VARCHAR(100) NOT NULL,

billetera VARCHAR(100) NOT NULL,

PRIMARY KEY (idPago),

FOREIGN KEY (idPago) REFERENCES Pago(idPago)

);

CREATE TABLE IF NOT EXISTS PagoMovil (

idPago INT,

referencia VARCHAR(100) NOT NULL,

telefono VARCHAR(20) NOT NULL,

PRIMARY KEY (idPago),

FOREIGN KEY (idPago) REFERENCES Pago(idPago)

);

CREATE TABLE IF NOT EXISTS Transferencia (

idPago INT,

bancoDestino VARCHAR(50) NOT NULL,

referencia VARCHAR(100) NOT NULL,

PRIMARY KEY (idPago),

FOREIGN KEY (idPago) REFERENCES Pago(idPago)

);

CREATE TABLE IF NOT EXISTS BioPago (

idPago INT,

referencia VARCHAR(100) NOT NULL,

PRIMARY KEY (idPago),

FOREIGN KEY (idPago) REFERENCES Pago(idPago)

);

CREATE TABLE IF NOT EXISTS Punto (

idPago INT,

punto INT NOT NULL,

PRIMARY KEY (idPago),

FOREIGN KEY (idPago) REFERENCES Pago(idPago)

);

CREATE TABLE IF NOT EXISTS Estadistica (

idEstadistica INT AUTO\_INCREMENT PRIMARY KEY,

producto1 INT,

cantidad1 INT NOT NULL,

producto2 INT,

cantidad2 INT NOT NULL,

producto3 INT,

cantidad3 INT NOT NULL,

cantidadOtros INT NOT NULL,

InicioMes DATE NOT NULL,

FinMes DATE NOT NULL,

FOREIGN KEY (producto1) REFERENCES Producto(idProducto),

FOREIGN KEY (producto2) REFERENCES Producto(idProducto),

FOREIGN KEY (producto3) REFERENCES Producto(idProducto)

);