

**Centro de Enseñanza Técnica Industrial**

**Desarrollo de Software**

**Practica 2.6 – Sistema de Facturación**

**Jesús Alberto Aréchiga Carrillo**

**22310439 4N**

**Profesor**

**José Luis García Cerpas**

**Mayo 2024**

**Guadalajara, Jalisco**

**Introducción**

Un sistema de facturación es un conjunto de procesos y herramientas utilizadas por las empresas para generar, enviar y administrar facturas por los bienes y servicios que ofrecen a sus clientes. Este sistema automatiza tareas como el cálculo de impuestos, el seguimiento de pagos y la generación de reportes financieros.

Es importante porque facilita la gestión financiera al registrar de manera precisa las transacciones comerciales, lo que ayuda a garantizar el cumplimiento de las obligaciones fiscales y a mantener un registro ordenado de los ingresos y gastos. Además, agiliza los procesos de facturación y cobro, lo que mejora la eficiencia operativa y la satisfacción del cliente.

**Desarrollo**

Primero hay que definir el diagrama relacional para poder hacer la base de datos física:

A screenshot of a computer

Description automatically generated

Ya teniendo el diagrama relacional, es posible hacer la base de datos física con sus respectivas relaciones.

Creación de la base de datos:

drop database if exists facturacion;

create database facturacion;

use facturacion;

create table if not exists categorias (

id int primary key auto\_increment,

nombre varchar(60)

);

create table if not exists productos (

id int not null auto\_increment,

nombre varchar(60) null default null,

precio decimal(10,0) null default null,

unidadMedida varchar(60) null default null,

id\_categoria int,

stock int,

stock\_min int,

descuento int,

foreign key(id\_categoria) references categorias(id),

primary key (id)

);

create table if not exists modo\_pago(

id int primary key,

nombre varchar(60),

detalles varchar (60)

);

create table if not exists clientes(

id int not null default '0',

nombre varchar(60) null default null,

direccion varchar(60) null default null,

rfc varchar(15) null default null,

modo\_pago int,

primary key (id),

foreign key (modo\_pago) references modo\_pago(id)

);

create table if not exists facturas(

id int not null auto\_increment,

fecha date null default null,

id\_cliente int null default null,

subtotal decimal(10,0) null default null,

notas varchar(60) default null,

foreign key (id\_cliente) references clientes(id)

on delete cascade,

primary key (id)

);

create table if not exists detalle\_fac(

id int not null auto\_increment,

id\_factura int null default null,

id\_producto int null default null,

cantidad int null default null,

foreign key (id\_factura) references facturas(id)

on delete cascade,

foreign key (id\_producto) references productos(id)

on delete cascade,

primary key(id)

);

insert into categorias(id, nombre) values (1, 'Herramienta Manual'),

(2, 'Desarmadores');

insert into modo\_pago(id, nombre, detalles) values (1, '4189334533345', 'Debito'),

(2, '123445677', 'Credito');

insert into clientes(id, nombre, direccion, rfc, modo\_pago) values (1, 'Juan', 'conocido', 'aaax6666', 1),

(2, 'Pedro', 'conocido', 'aaax6666', 2),

(3, 'Oscar', 'conocido', 'aaax6666', 2);

insert into productos(id, nombre, precio, unidadMedida, id\_categoria, stock, stock\_min, descuento) values

(1, 'Mazo', 90, 'pieza', 1, 100, 100, 60),

(2, 'Pinza', 75, 'pieza', 1, 50, 100, 60),

(3, 'Desarmador', 100, 'pieza', 2, 80, 100, 60),

(4, 'Martillo', 50, 'pieza', 1, 50, 20, 10),

(5, 'Taladro', 89, 'pieza', 1, 10, 10, 20);

Al tener este script, es posible empezar a hacer las operaciones deseadas:

Primero se crean facturas para después poder ingresar el detalle de la factura, agregando el producto.

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

De esta manera se pueden ir ingresando todos los datos necesarios para completar todas las facturas.

Para obtener la información, es necesario