

Proyecto SQL Cristina`s Empanadas

Introducción:

El proyecto consiste en que la base de datos que voy a ir creando pueda cubrir todo lo necesario de un ecommerce, una página de venta de empanadas. Este proyecto es una continuación del proyecto que hice en javascript en Coder House.

Situación problemática:

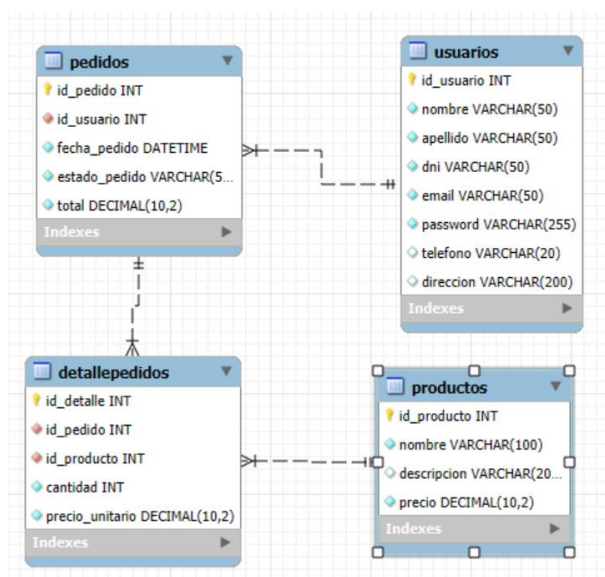
La situación que resolvería la base de datos es todo el proceso de compra de productos, generación de pedidos y creación de usuarios.

Modelo de negocio:

Necesito tener constancia de todas las compras que se realizan para la contabilidad del negocio (ganancias, perdidas), necesito saber todos los pedidos que se generan para la producción de empanadas, y necesito tener todos los usuarios creados y la posibilidad de agregar nuevos usuarios.

El usuario puede loguearse o crear un nuevo usuario, luego puede ver todas las opciones de empanadas, genera el pedido que desea y realiza la compra.

Diagrama Entidad-Relación:



Listado de tablas:

- Usuarios

Esta tabla contiene la información de cada cliente.

- `id_usuario`: `INT` - Clave Primaria
- `nombre`: `VARCHAR(50)`
- `apellido`: `VARCHAR(50)`
- `dni`: `VARCHAR(50)` - Clave Única
- `email`: `VARCHAR(50)` - Clave Única
- `password`: `VARCHAR(255)`
- `telefono`: `VARCHAR(20)`
- `direccion`: `VARCHAR(200)`

- Productos

Esta tabla almacena los datos de cada producto que vendes.

- `id_producto`: `INT` - Clave Primaria
- `nombre`: `VARCHAR(100)` - Clave Única
- `descripcion`: `TEXT`
- `precio`: `DECIMAL(10, 2)`

- Pedidos

Esta tabla registra cada orden de compra realizada.

- `id_pedido`: `INT` - Clave Primaria
- `id_usuario`: `INT` - Clave Foránea (referencia a `Usuarios`).
- `fecha_pedido`: `DATETIME`
- `estado_pedido`: `VARCHAR(50)`
- `total`: `DECIMAL(10, 2)`

DetallePedidos

Esta tabla es el conector que especifica los productos y sus cantidades en cada pedido.

- `id_detalle`: `INT` - ****Clave Primaria****.
- `id_pedido`: `INT` - ****Clave Foránea**** (referencia a `Pedidos`).
- `id_producto`: `INT` - ****Clave Foránea**** (referencia a `Productos`).
- `cantidad`: `INT`
- `precio_unitario`: `DECIMAL(10, 2)`

Listado de vistas:

- historialdepedidos

Esta vista nos da un resumen de todos los pedidos que se realizaron, con el objetivo de poder controlar cada estado de pedido.

- rankingproductos

Esta vista nos dará el ranking de los productos más vendidos, esto nos sirve con el objetivo de saber cuáles son los productos más deseados y poder gestionar el inventario a futuro.

- resumenventasdiarias

Esta vista nos dará como resultado la cantidad de pedidos y de ganancia por día, esto tiene como objetivo ver la contabilidad, ganancia o facturación del negocio.

Listado de funciones:

- `calcularpreciocondescuento`

Esta función sirve para hacer el descuento que podría haber en el negocio al precio final.

- `descripcionestadopedido`

Esta función nos devolvería una descripción del estado del pedido para poder comunicarle al cliente en qué situación se encuentra su pedido. Por ejemplo: "Completado" significa que el pedido ya se entregó.

Listado de procedimientos almacenados:

`obtenerdetallepedido`

Nos devuelve el detalle, es decir los productos, que contienen un pedido en particular. Tiene como objetivo verificar que productos deben ser enviados al cliente o que productos deben estar en la orden.

`obtenerpedidosactivos`

Nos devuelve todos los pedidos que no fueron entregados. Tiene como objetivo poder verificar que pedidos nos faltan entregar y ajustar lo que requiera para que sean completados.

`actualizarestadopedido`

Nos da la posibilidad de actualizar el estado de un pedido. Por ejemplo, si se completa un pedido, pasará de "Pendiente" a "Completado".

Listado de triggers:

tr_pedidos_insert

Este trigger deja registrado todos los logs de inserción a la tabla de pedidos. En la tabla pedidos_log.

tr_pedidos_Update

Este trigger deja registrado todos los logs de actualización a la tabla de pedidos. En la tabla pedidos_log.

tr_pedidos_Delete

Este trigger deja registrado todos los logs de eliminación a la tabla de pedidos. En la tabla pedidos_log.

Todos estos triggers sirven para una auditoría de la creación, eliminación y actualización de los pedidos. Sirve para saber si los empleados o responsables hacen bien la carga de los pedidos.