



PROYECTO FINAL CURSO SQL DE CODERHOUSE. E-COMMERCE

Documentación técnica del proyecto.

Alumno: María Luján Benjamín

Profesor: Camilo Redondo

Comisión: 57185

INTRODUCCION.

Este proyecto de base de datos para un sistema de e-commerce fue planteado como proyecto final de la cursada 57185. Del curso de Coderhouse SQL a cargo del profesor Camilo Redondo. Tiene como objetivo demostrar lo aprendido durante el curso que tuvo una duración de 50 horas reloj. A continuación se detalla el objetivo de la base de datos, descripción de la base y las partes que se incorporaron más una conclusión del proyecto.

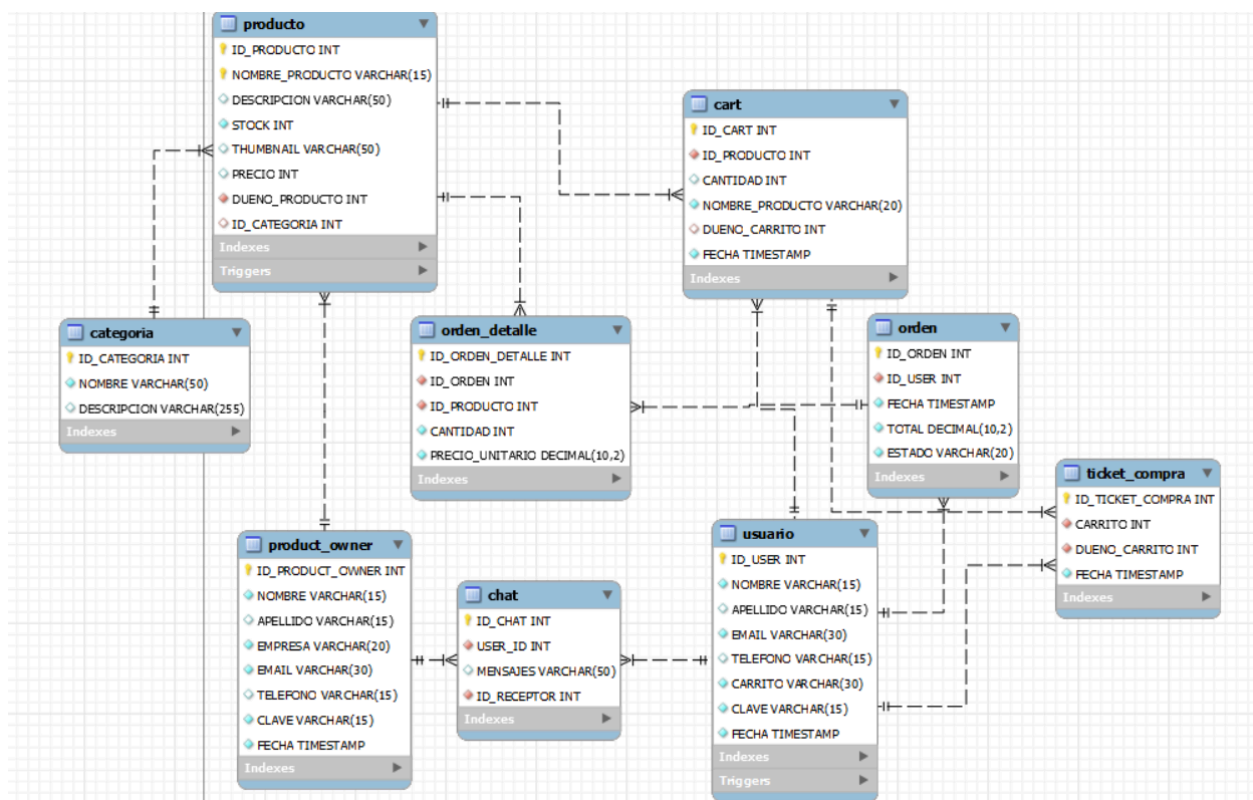
OBJETIVO.

El objetivo principal de la base de datos para este sistema de e-commerce es proporcionar una infraestructura eficiente que facilite la gestión integral de la tienda en línea, garantizando la seguridad de los datos, la eficiencia en la administración de inventarios, la mejora de la experiencia del usuario y la transparencia en las transacciones. Este objetivo se desglosa en los siguientes puntos:

1. **Seguridad y Protección de Datos:** Asegurar que la información personal y las credenciales de los usuarios estén protegidas contra accesos no autorizados mediante la implementación de restricciones.
2. **Eficiencia en la Gestión de Productos e Inventario:** Permitir una organización clara y accesible de los productos a través de una estructura categorizada, facilitando la búsqueda y selección de artículos por parte de los usuarios y asegurando la precisión en la gestión del inventario.
3. **Optimización de la Experiencia del Usuario:** Proporcionar a los usuarios una experiencia de compra fluida mediante la gestión eficiente de carritos de compra dinámicos y la actualización en tiempo real del inventario y precios.
4. **Facilitación de la Comunicación:** Implementar un sistema de chat eficaz que permita la comunicación directa entre compradores y vendedores, mejorando la atención al cliente y resolviendo rápidamente cualquier duda o problema.

5. **Transparencia y Registro de Transacciones:** Llevar un registro detallado y confiable de todas las transacciones y órdenes, proporcionando transparencia en el proceso de compra y facilitando la gestión contable de la tienda.
6. **Escalabilidad y Flexibilidad:** Diseñar una estructura de base de datos que sea escalable y flexible, permitiendo futuras expansiones y adaptaciones a medida que la tienda crezca o cambien las necesidades del negocio.

ESTRUCTURA DE LA BASE DE DATOS (DER)



ESQUEMA DE LA BASE DE DATOS.

Tabla	Campo	Descripción	Tipo De Dato
Usuario	ID_USER	identificación única del usuario	número
	NOMBRE	nombre del usuario	texto
	APELLIDO	apellido del usuario	texto
	E-MAIL	email de contacto del usuario	texto
	TELEFONO	teléfono del usuario	número
	CARRITO	carrito de compra del usuario	número
	CLAVE	clave de acceso del usuario	alfanumerico
	FECHA	fecha de registro del usuario	número
Product_owner	ID_PRODUCT_OWNER	identificación única del que publica el producto	número
	NOMBRE	nombre del propietario del producto	texto
	APELLIDO	apellido del propietario del producto	texto
	EMPRESA	empresa para la que trabaja	texto
	EMAIL	email de contacto del propietario del producto	texto
	TELEFONO	teléfono de contacto	numero
	CLAVE	clave de acceso a la plataforma	alfanumerico
	FECHA	fecha de registro del propietario del productos	número
Categoria	ID_CATEGORIA	número de identificación de la categoría	número
	NOMBRE	nombre de la categoría a la que pertenece el producto	texto
	DESCRIPCION	descripción de la categoría	texto
Producto	ID_PRODUCTO	número de identificación del producto	número
	NOMBRE_PRODUCTO	nombre del producto que se publica	texto
	DESCRIPCION	descripción del producto	texto
	STOCK	cantidad de producto que se publica	número
	THUMBNAIL	link de la imagen del producto	texto
	PRECIO	precio al que se va a vender el producto	número
	DUENO_PRODUCTO	propietario del producto/quien lo publica	número

	ID_CATEGORIA	categoría a la que pertenece el producto	número
Cart	ID_CART	número de identificación de carrito de compra	número
	ID_PRODUCTO	número de identificación del producto que se compra	número
	CANTIDAD	cantidad de producto que se compra	número
	NOMBRE_PRODUCTO	nombre del producto que se compra	texto
	DUENO_CARRITO	a qué usuario pertenece el carrito	número
	FECHA	fecha de compra	número
Chat	ID_CHAT	número de identificación del chat que se inició	número
	ID_USER	identificación del emisor del mensaje	número
	MENSAJE	mensaje que se comunica	texto
	ID_RECEPTOR	identificación del receptor del mensaje	número
Ticket_compra	ID_TICKET_COMPRA	identificación del fin de compra	número
	CARRITO	carrito al que pertenece esta compra	número
	DUENO_CARRITO	dueño del carrito comprado	número
	FECHA	fecha de la compra	número
Orden	ID_ORDEN	identificación de la orden de compra	número
	ID_USER	identificación del usuario que hizo la orden	número
	FECHA	fecha en la que se realizó la orden	número
	TOTAL	total que se gastó en esta orden	número
	ESTADO	estado del pedido	texto
Orden_detalle	ID_ORDEN_DETALLE	identificación del detalle de la orden	número
	ID_ORDEN	número de identificación de la orden	número
	ID_PRODUCTO	número de identificación del producto	número
	CANTIDAD	cantidad que se pidió del producto	número
	PRECIO_UNITARIO	precio unitario del producto que se pidió	número

VISTAS.

Esta base cuenta con una vista en la base principal, para trabajar y proteger los datos propios antes de modificarlos. vw_productos y el resto en un schema aparte para uso y trabajo de otro equipo.

VW_PRODUCTOS: Muestra todos los productos tal como están en la tabla de productos.

VW_PRODUCTOS_DUENOS_Y_SU_CATEGORIA: Muestra a los propietarios de los productos con sus productos y las categorías a las que pertenecen, incluso si no tienen productos.

VW_3_PRODUCTOS_MAS_COMPRADOS: Muestra los tres productos que más se han comprado.

VW_PRODUCTOS_MAS_COMPRADOS_POR_USUARIO: Muestra los productos que los usuarios han comprado, con información del usuario, la fecha de la compra, la cantidad y el costo total.

VW_2_USUARIOS_CON_MAS_COMPRARAS: Muestra los dos usuarios que más han gastado en compras, con el total que han gastado.
VW_CARRITOS_COMPRADOS_HOY: Muestra los carritos de compras que se han completado hoy.

FUNCIONES.

FN_PROMEDIO_PRECIO: Calcula el precio promedio de los productos en una categoría específica.

FN_PRODUCTO_MAS_VENDIDO: Encuentra cuál fue el producto más vendido entre dos fechas específicas.

PROCEDIMIENTOS.

SP_ORDENAMIENTO: Permite ordenar cualquier tabla según una columna específica en orden ascendente o descendente, y guarda el resultado en una tabla temporal.

SP_INSERTAR_USUARIO: Agrega un nuevo usuario a la tabla de usuarios con la información proporcionada, incluyendo nombre, apellido, correo electrónico, teléfono, carrito y clave.

DISPARADORES.

TRG_INSERT_PRODUCTO: Después de que se agrega un nuevo producto, guarda en una tabla de auditoría el tipo de acción (insertar), el nombre de la tabla, el ID del producto, el usuario que realizó la acción y la fecha y hora de la acción.

trigger_update_usuario: Antes de que se actualice un usuario, guarda en una tabla de auditoría el tipo de acción (actualizar), el usuario que realizó la actualización, el ID del usuario que se está modificando, el correo electrónico del usuario antes de la actualización, y la fecha y hora de la acción.


USUARIOS.

Se crean 2 usuarios nuevos de la base con dos permisos diferentes

Creación del Usuario 1:

Se crea un nuevo usuario con una contraseña. Se muestra qué permisos tiene este usuario. Se le otorga permiso para ver la información de la tabla ORDEN_DETALLE. Para poder ver la información de el detalle de una compra

Creación del Usuario 2:



Se crea otro nuevo usuario con una contraseña diferente. Se le otorgan permisos para ver, agregar y modificar información en la tabla PRODUCTO.

Archivos de prueba: Hay dos archivos SQL que se utilizan para probar si los usuarios fueron creados correctamente y tienen los permisos adecuados.

TRANSACCIONES, EJEMPLO DE USO.

El proyecto también cuenta con un pequeño módulo de prueba de transacciones donde,

Cambio de Configuración: Cambia algunas configuraciones para hacer pruebas sin guardar automáticamente los cambios y sin verificar relaciones entre tablas.

Tabla 1: Propietarios de Productos:

Muestra todos los datos de la tabla de propietarios de productos.

Inicia una transacción y borra tres propietarios específicos.

Verifica si los propietarios fueron eliminados.

Tabla 2: Productos:

Muestra todos los datos de la tabla de productos.

Inicia una transacción e inserta ocho nuevos productos.

Crea dos puntos de guardado (para volver a estos puntos si es necesario).

Finalización de la Transacción:

Puede decidirse entre deshacer los cambios o confirmarlos.

CONCLUSION.

Este proyecto de base de datos para un sistema de e-commerce busca resolver problemas comunes de las tiendas en línea. Con una estructura bien pensada, la base de datos asegura:

1. **Protección de Datos:** Los datos de los usuarios están seguros.
2. **Gestión Eficiente del Inventario:** Control preciso del stock.
3. **Mejora de la Experiencia del Usuario:** Compras fáciles y rápidas.
4. **Comunicación Directa:** Chat efectivo entre compradores y vendedores.
5. **Transparencia en las transacciones:** Registros claros de todas las compras.

Además, la base de datos puede crecer y adaptarse a futuras necesidades del negocio. Con la creación de usuarios con permisos específicos y el uso de transacciones seguras, el sistema es robusto y eficiente. En resumen, este proyecto proporciona una base sólida para una tienda en línea exitosa y en crecimiento.