



Fabián Paredes

Go E-commerce (Sistemas de Gestión Empresarial)

Contenido

Go E-commerce (Sistemas de Gestión Empresarial).....	3
Introducción	3
Objetivos	3
Tecnologías usadas	3
Estructura del repositorio	4
Rutas principales	4
Notas sobre la base de datos	4
Configuración y ejecución	5
Links	5
Aprendizaje	5

Go E-commerce (Sistemas de Gestión Empresarial)

Introducción

Este proyecto es una aplicación de ejemplo de e-commerce escrita en Go. Implementa funcionalidades básicas de tienda online: listado y detalle de productos, carrito de compras, proceso de checkout (simulado), gestión de pedidos y panel administrativo para productos, pedidos y clientes.

Objetivos

- Proveer una aplicación didáctica que muestre conceptos de web en Go.
- Practicar acceso a base de datos MySQL, plantillas HTML y manejo de sesiones mediante cookies.
- Servir como base para extender funcionalidades (autenticación segura, pasarelas de pago, API REST).

Tecnologías usadas

- Lenguaje: Go
- Router: Gorilla Mux
- Plantillas: Go html/template
- Base de datos: MySQL (con github.com/go-sql-driver/mysql)
- Variables de entorno: github.com/joho/godotenv

Estructura del repositorio

- eCommerce.go — Punto de entrada y registro de rutas.
- db/ — Conexión a la base de datos (conexion.go).
- handlers/ — Controladores HTTP para cliente y administrador.
- models/ — Lógica de acceso a datos (productos, clientes, carrito, pedidos).
- templates/ — Plantillas HTML (layouts, vistas cliente/admin).
- static/ — Archivos estáticos (CSS, JS, imágenes).

Rutas principales

- / — Página principal (GET)
- /login, /register — Login y registro (GET/POST)
- /producto/{id} — Detalle de producto (GET)
- /carrito — Ver carrito (GET)
- /producto/agregar-carrito — Añadir al carrito (POST)
- /checkout — Checkout (GET/POST)
- /perfil — Perfil y lista de pedidos (GET)
- /admin/* — Panel administrativo (productos, pedidos, clientes)

Notas sobre la base de datos

Se asume un esquema con

tablas: clientes, productos, carritos, items_carrito, pedidos, detalles_pedido, y tablas

auxiliares. Las consultas SQL están embebidas en el código y esperan columnas con nombres específicos (p.ej. id_producto, id_cliente, total).

Configuración y ejecución

1. Crear archivo. env con las variables:

```
DB_USER=root
DB_PASSWORD=tu_password
DB_HOST=127.0.0.1
DB_PORT=3306
DB_NAME=nombre_basedatos
PORT=8080
```

2. Descargar dependencias:
go mod tidy
3. Ejecutar la aplicación:
go run ./ecommerce.go

Links

- [Video](#)
- [Github](#)

Aprendizaje

En este proyecto se consolidaron conocimientos prácticos sobre desarrollo web en Go: diseño de rutas con Gorilla Mux, renderizado de vistas con `html/template`, acceso y manipulación de datos en MySQL desde modelos, y manejo básico de sesiones mediante cookies. También se adquirió experiencia útil en organización de un proyecto (separación `handlers/models/db`), en detección y manejo de errores en tiempo de ejecución (logs y respuestas HTTP). En conjunto, el proyecto sirvió como laboratorio para integrar componentes backend, aprender a depurar plantillas y consultas SQL, y sentar las bases para futuras mejoras y pruebas automatizadas.