



UNIVERSIDAD DR. ANDRÉS BELLO

Facultad de Tecnología e Innovación

Regional Chalatenango

Tema:

Configuración de API REST

Docente:

Jose Rosalio Serrano Carvajal

Presentado por

Autores:
Arnulfo Josué Santos Jiménez
Carlos Alejandro Núñez Leiva
Eduardo Antonio Vásquez López
Kevin Mauricio Alvarenga Henríquez

Fecha:

Chalatenango, 22 de julio de 2025

Configuración API

Go (Goland)

Esta es una API desarrollada en Go (Golang) que simula un backend para una tienda en línea de productos tecnológicos. Permite realizar operaciones CRUD sobre productos del inventario.

Características

- Consultar todos los productos
- Consultar un producto por ID
- Agregar un nuevo producto
- Editar un producto
- Eliminar un producto

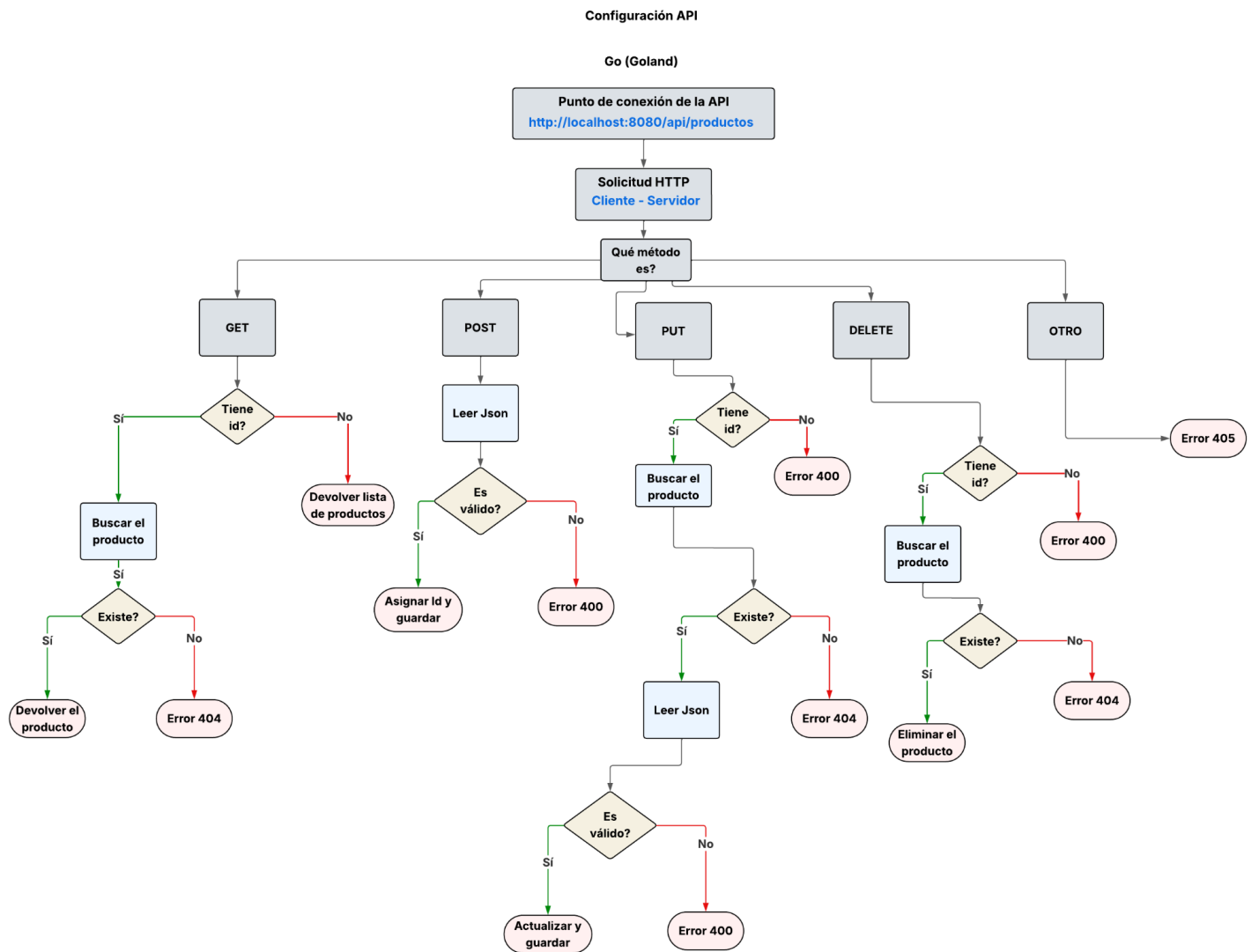
Estructura API

```
productos-api/
├── go.mod
├── main.go          // Arranca el servidor y se configuran las rutas
|
├── handlers/        // Lógica de manejo de rutas y peticiones
|   └── producto_handler.go
|
├── models/          // Definición de estructuras de los datos
|   └── producto.go
|
├── data/            // "Base de datos" temporal (lista en memoria)
|   └── producto_data.go
|
└── README.md        // Explicación del proyecto
```

Métodos utilizados

Método	Ruta	Acción
Get	/api/productos	Obtiene todos los productos
Get	/api/productos?id=1	Obtiene un producto específico
Post	/api/productos	Crea un nuevo producto
Put	/api/productos?id=1	Actualiza un producto existente
Delete	/api/productos?id=1	Elimina un producto

Diagrama



Ejecución de la API

1. Clonar el repositorio:

- `git clone https://github.com/tu-usuario/productos-api.git`
- `cd productos-api`

2. Abrir Archivo en Visual Studio Code: Abre la terminal y ejecutar la API:

- `go run main.go`
- Servidor corriendo en `http://localhost:8080`

3. Abrir Postman y crear nueva solicitud: Crear un producto (POST):

- En Postman, hacer clic en "New" > HTTP Request
- Método: POST
- URL: `http://localhost:8080/api/productos`
- Ir a la pestaña "Body", selecciona "raw" y seleccionar tipo JSON.

Escribir algo como:

```
{  
  "nombre": "Mouse",  
  "descripcion": "Gamer",  
  "precio": 25,  
  "stock": 10  
}
```

4. Ver todos los productos (GET)

- Método: GET
- URL: `http://localhost:8080/api/productos`
- Hacer clic en "Send"

5. Actualizar un producto (PUT)

- Método: PUT
- URL: `http://localhost:8080/api/productos?id=1` (Cambiar id por el que se desea actualizar)
- Ir a Body > raw > JSON y escribir algo como:

```
{  
  "nombre": "Mouse Pro XL",  
  "descripcion": "Gamer",  
  "precio": 30,  
  "stock": 10  
}
```
- Hacer clic en "Send"
- saldrá el mensaje: "Producto actualizado"

6. Eliminar un producto (DELETE)

- Método: DELETE
- URL: `http://localhost:8080/api/productos?id=1` (Cambiar id por el que se desea eliminar)
- Haz clic en "Send"
- saldrá el mensaje: "Producto eliminado"