

# Diagrama de Infraestructura - Microservicio Tareas (Golang + MongoDB)

## Componentes del Sistema

### 1. Cliente / Entorno local

- Thunder Client (VSCode) para pruebas.
- Navegador o cliente HTTP.

### 2. Contenedor: Microservicio (Go)

- Imagen construida con Dockerfile (Multi-stage Build).
- Expone el puerto 8080.
- Accede a variables de entorno definidas en .env:
  - MONGO\_URI
  - MONGO\_DB
  - PORT
  - ENV
- La URI cambia según el entorno:
  - ENV=dev → MONGO\_URI\_TEST=mongodb://localhost:27017
  - ENV=prod → MONGO\_URI=mongodb://mongo:27017

### 3. Contenedor: MongoDB

- Imagen oficial mongo :6.0.
- Puerto expuesto: 27017.
- Volumen persistente mongodb\_data para conservar datos.
- Se genera respaldo con mongodump y se guarda dentro del contenedor.

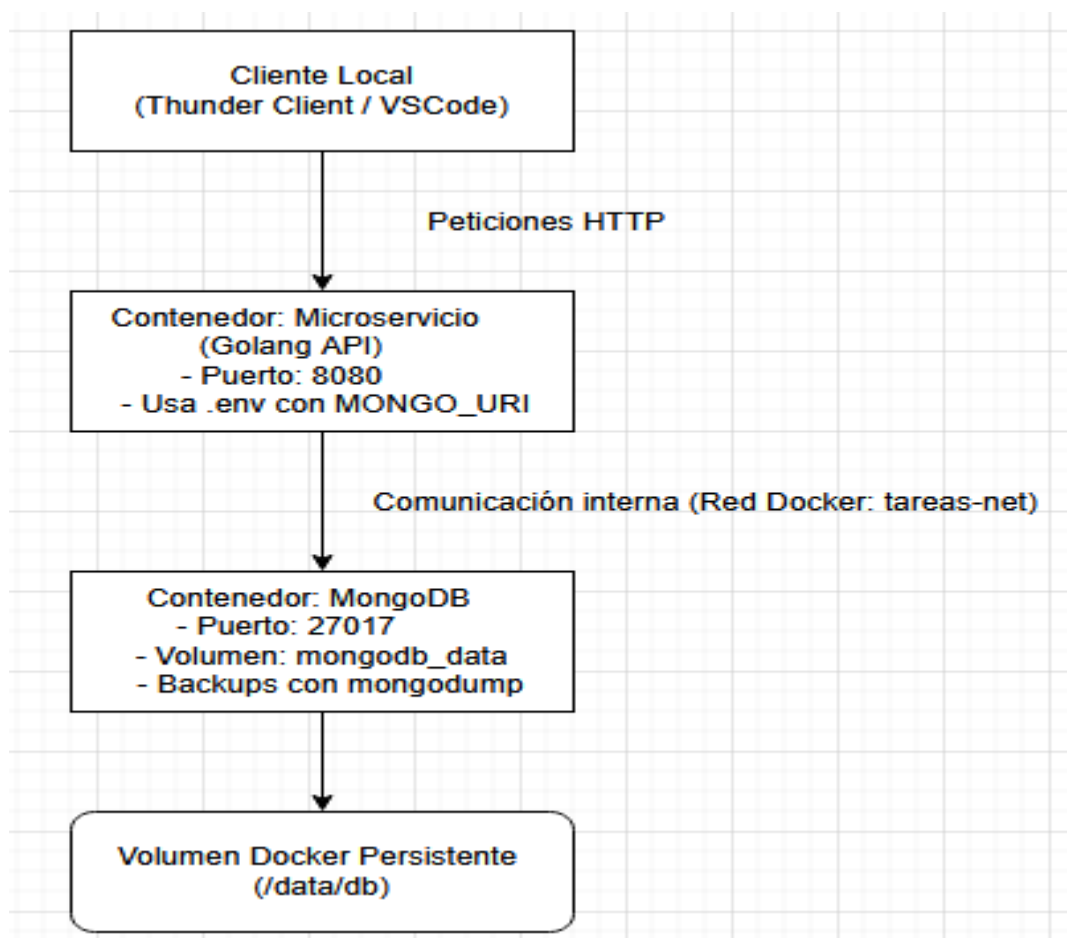
#### 4. Red Docker personalizada: tareas-net

- Ambos contenedores (Go + MongoDB) se comunican internamente por esta red.

#### 5. Volumen Docker: mongodb\_data

- Asociado a **/data/db** dentro del contenedor de MongoDB.
- Permite persistencia de datos entre reinicios.

### Diagrama Visual (esquemático)



### Seguridad

- MongoDB no expone el puerto 27017 al exterior en producción (solo dentro de la red Docker).
- Acceso controlado por variables de entorno y redes Docker internas.

