Conversación sobre Kubernetes, Pods, Services y Red

Esta conversación trata sobre conceptos fundamentales de Kubernetes, como Pods, Containers, Network Namespaces, comunicación entre servicios, uso de DNS internos, exposición de servicios, y la arquitectura de red dentro de un clúster local usando Docker Desktop con Kubernetes habilitado.

## ¿Qué es una pila de red y/o un network namespace?

La pila de red es el conjunto de componentes del sistema operativo que permiten comunicación por red. Incluye interfaces, IPs, puertos, reglas de firewall y protocolos. Un network namespace es una copia aislada de esa pila de red: tiene sus propias interfaces, IPs y espacio de puertos. Docker y Kubernetes usan namespaces para aislar redes entre contenedores o pods.

## ¿Cómo puede un frontend acceder al backend en Kubernetes si están en pods distintos?

Debe usar el nombre de un Service (ej. `backend-svc`) que apunte al backend. Kubernetes tiene un DNS interno que permite a los pods comunicarse usando nombres como `http://backend-svc:3000`, y no `localhost`, ya que cada pod tiene su propia IP.

## ¿Puede el frontend acceder directamente al pod del backend por su DNS?

Sí, cada pod tiene un DNS interno como `backend-pod.default.pod.cluster.local`. Pero no es recomendable usarlo directamente porque los pods pueden reiniciarse, cambiar de nombre o IP. En cambio, usar un Service como `backend-svc` permite balanceo de carga y estabilidad.

## ¿Un pod se expone mediante un Service o solo sus contenedores internos?

Un Service expone uno o más pods (no contenedores individuales). Dentro del pod, los contenedores comparten IP y puertos. El Service se conecta al puerto del pod que esté escuchando, lo cual normalmente corresponde al contenedor principal.