Desafío 11 – Kubernetes Deployment

# 1. Objetivo

El objetivo de este desafío es poner en práctica lo aprendido sobre Kubernetes, desplegando una aplicación NestJS junto con una base de datos MongoDB en un entorno de clúster. Este desafío es una evolución del trabajo realizado en el Desafío 10, donde se utilizó Docker Compose para levantar los servicios.

# 2. Descripción General

El proyecto consiste en transformar los servicios definidos anteriormente en un conjunto de manifiestos YAML de Kubernetes, que describen el despliegue de la aplicación y de la base de datos MongoDB. Esto incluye los objetos Deployment, Service y PersistentVolumeClaim.

# 3. Estructura del Proyecto

La estructura del proyecto es la siguiente:

desafio\_11/  
├── manifests/  
│ ├── app-deployment.yaml  
│ ├── app-service.yaml  
│ ├── mongo-deployment.yaml  
│ ├── mongo-service.yaml  
│ └── mongo-pvc.yaml  
├── documentacion\_desafio\_11.docx  
├── evidencia/  
│ └── captura\_kubectl\_get\_pods.png  
└── diagrama.png

# 4. Componentes Desplegados

• app-deployment.yaml: despliega la aplicación NestJS.  
• app-service.yaml: expone la aplicación a través de un servicio ClusterIP.  
• mongo-deployment.yaml: despliega una instancia de MongoDB.  
• mongo-service.yaml: permite que la aplicación acceda a la base de datos mediante un nombre DNS interno.  
• mongo-pvc.yaml: define un volumen persistente para la base de datos.

# 5. Requisitos Técnicos

• Tener instalado minikube o acceso a un clúster de Kubernetes.  
• Tener kubectl configurado y funcionando.  
• Docker para construir la imagen de la aplicación.

# 6. Mejoras Potenciales

• Incorporar Helm Charts para facilitar la gestión del despliegue.  
• Integrar GitHub Actions para CI/CD.  
• Utilizar Ingress para exponer la aplicación fuera del clúster.  
• Añadir monitoreo con Prometheus y Grafana.

# 7. Evidencia de Ejecución

Se adjunta en el repositorio el archivo `kubectl\_output.txt`.

# 8. Diagrama

