



## **Sistema de Monitoreo de Almacén Automatizado con Kubernetes**

Juan Alberto Guillen Albarracin

Sistemas distribuidos

Cristian Timbi

## 1. Introducción

Este documento presenta el diseño, implementación y validación de un sistema de monitoreo en tiempo real para un almacén automatizado. El sistema se despliega sobre Kubernetes y demuestra persistencia de datos, seguridad básica y resiliencia ante fallos.

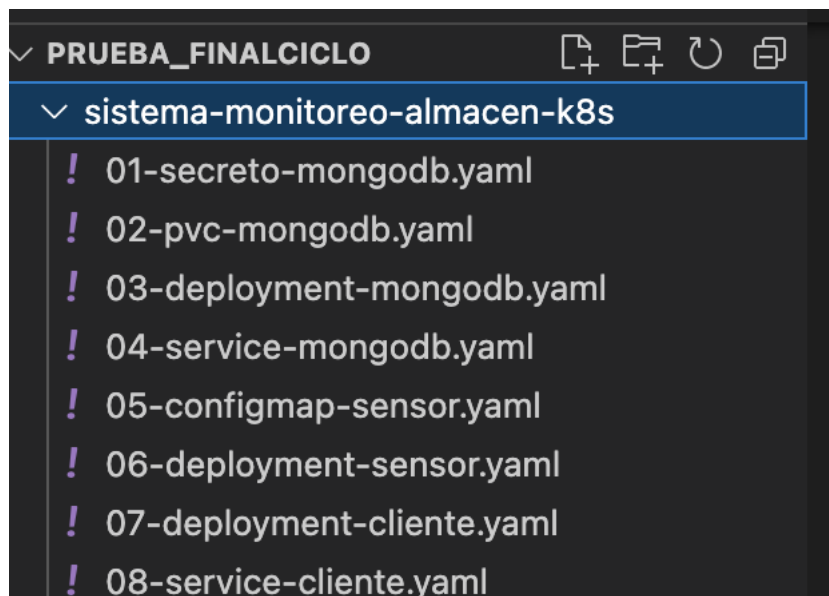
## 2. Tabla de Tecnologías Seleccionadas

Componente	Tecnología	Justificación
Orquestación	Kubernetes	Gestión de contenedores y resiliencia
Persistencia	MongoDB	Base NoSQL orientada a documentos JSON
Almacenamiento	PersistentVolumeClaim	Persistencia de datos ante fallos
Productor	Bash + mongosh	Generación simple y continua de datos
Visualización	Mongo Express	Interfaz web existente para MongoDB

## 3. Manifiestos de Kubernetes

Se implementaron manifiestos Kubernetes para Secrets, PersistentVolumeClaim, Deployments y Services. El script del microservicio productor se definió en un ConfigMap y se revisó especialmente la sintaxis de conexión autenticada a la base de datos.

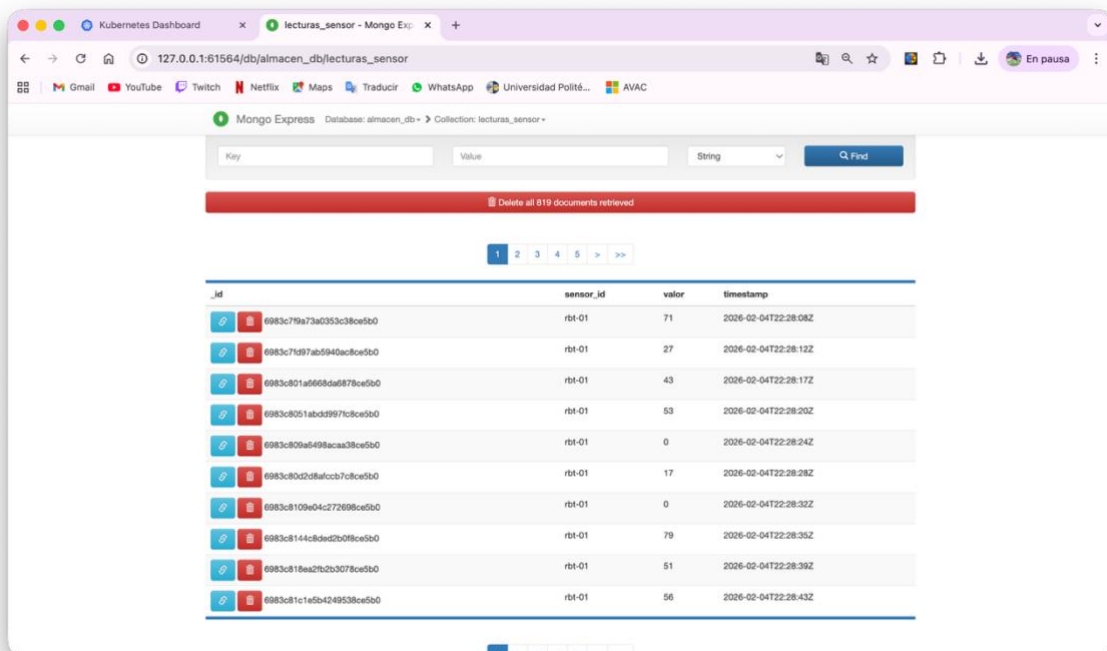
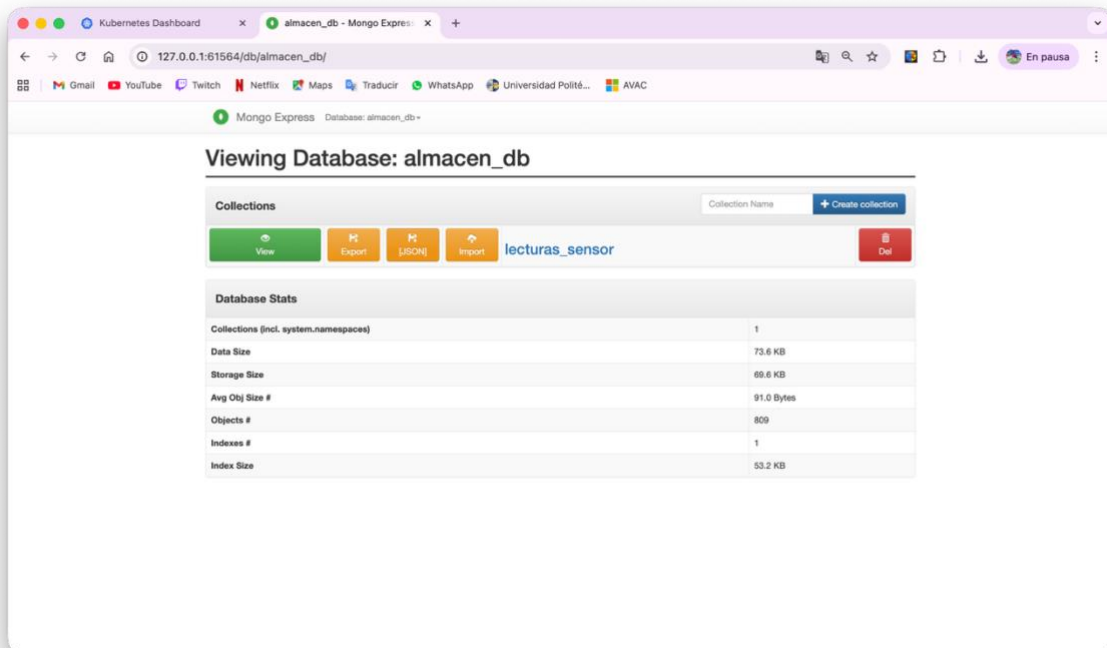
(Los archivos YAML completos se incluyen en el repositorio Git)



## 4. Evidencias de Funcionamiento

### 4.1 Interfaz Web de Visualización

Captura de la interfaz web (Mongo Express) mostrando los datos almacenados con el formato JSON solicitado.



## 4.2 Persistencia del Volumen

Evidencia de consola que muestra el estado del PersistentVolumeClaim en condición Bound.

```
(base) juanguillenalbarracin@Juans-MacBook-Pro sistema-monitoreo-almacen-k8s % kubectl get pvc
```

NAME	STATUS	VOLUME	CAPACITY	ACCESS MODES	STORAGECLASS	VOLUMEATTRIBUTESCLASS
pvc-mongodb	Bound	pvc-bd8a91f9-0cf8-4605-adee-b94784230c7c	1Gi	RWO	standard	<unset>

```
(base) juanguillenalbarracin@Juans-MacBook-Pro sistema-monitoreo-almacen-k8s %
```

## 5. Prueba de Resiliencia y Recuperación

Para validar la resiliencia del sistema, se eliminó manualmente el Pod de MongoDB. Kubernetes recreó automáticamente el Pod y los datos previamente almacenados permanecieron disponibles gracias al volumen persistente.

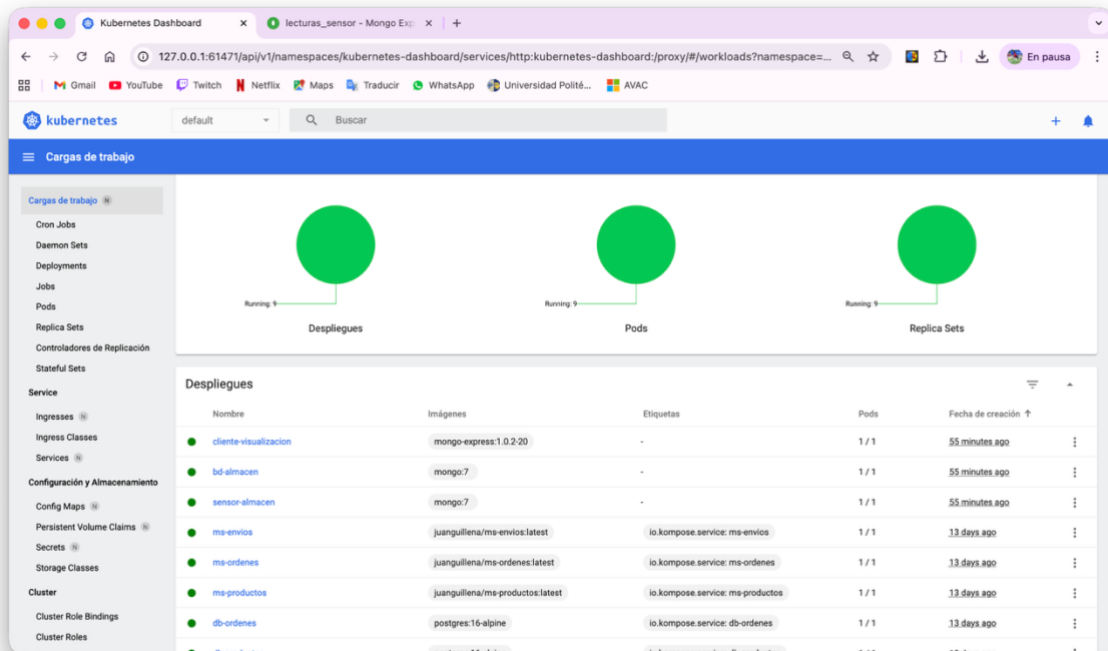
```
(base) juanguillenalbarracin@Juans-MacBook-Pro sistema-monitoreo-almacen-k8s % kubectl delete pod -l app=bd-almacen
pod "bd-almacen-564c9f49d5-jwhm9" deleted
```

```
(base) juanguillenalbarracin@Juans-MacBook-Pro sistema-monitoreo-almacen-k8s % kubectl get pods -w
```

NAME	READY	STATUS	RESTARTS	AGE
bd-almacen-564c9f49d5-r8tmr	1/1	Running	0	49s
cliente-visualizacion-854c8bfc55-jrxrt	1/1	Running	0	5m19s
db-ordenes-f6d576db-7ztn8	1/1	Running	1 (6m40s ago)	8d
db-productos-9f78f8dbf-r4h8d	1/1	Running	1 (6m40s ago)	8d
gateway-api-7b77bdf5f-cd6jn	1/1	Running	2	8d
ms-envios-d59959749-ljz2x	1/1	Running	0	8d
ms-ordenes-bc85dfbc8-hzjqz	1/1	Running	0	8d
ms-productos-77d9f867dc-wmgqn	1/1	Running	0	8d
sensor-almacen-f8ff6878d-28m4s	0/1	CrashLoopBackOff	3 (48s ago)	5m19s
sensor-almacen-f8ff6878d-28m4s	1/1	Running	4 (50s ago)	5m21s

NAME	READY	STATUS	RESTARTS	AGE
bd-almacen-564c9f49d5-r8tmr	1/1	Running	0	12s
cliente-visualizacion-854c8bfc55-jrxrt	1/1	Running	0	4m42s
db-ordenes-f6d576db-7ztn8	1/1	Running	1 (6m3s ago)	8d
db-productos-9f78f8dbf-r4h8d	1/1	Running	1 (6m3s ago)	8d
gateway-api-7b77bdf5f-cd6jn	1/1	Running	2	8d
ms-envios-d59959749-ljz2x	1/1	Running	0	8d
ms-ordenes-bc85dfbc8-hzjqz	1/1	Running	0	8d
ms-productos-77d9f867dc-wmgqn	1/1	Running	0	8d
sensor-almacen-f8ff6878d-28m4s	0/1	Error	3 (3m15s ago)	4m42s
sensor-almacen-f8ff6878d-28m4s	0/1	CrashLoopBackOff	3 (12s ago)	4m43s



## 6. Conclusiones

El proyecto cumple con todos los requerimientos establecidos, demostrando una arquitectura desacoplada, persistencia de datos, seguridad básica y recuperación automática ante fallos. La solución es adecuada para un entorno académico y permite comprender el funcionamiento de Kubernetes en sistemas distribuidos.

## 7. Tabla de Tecnologías Seleccionadas

Componente	Tecnología	Justificación
Orquestación	Kubernetes	Gestión de contenedores y resiliencia
Persistencia	MongoDB	Base NoSQL orientada a documentos JSON
Almacenamiento	PersistentVolumeClaim	Persistencia de datos ante fallos
Productor	Bash + mongosh	Generación simple y continua de datos
Visualización	Mongo Express	Interfaz web existente para MongoDB

