Integrantes

Quintana Cubas Lenin Alexander  
Ponte Arica Anthony Rosemberg  
Ramirez Vasquez Jonathan  
Silva Ramos Juan Armando

Docker y Kubernetes Intermedio

Proyecto Final del Curso

Contenido

[Proyecto Curso Docker y Kubernetes 2](#_Toc198593602)

[Entregables del Proyecto Final: Gestión de Eventos y Participantes 2](#_Toc198593603)

[1. Código Fuente 2](#_Toc198593604)

[1.1. Repositorios de Microservicios: 2](#_Toc198593605)

[2. Artefactos Contenerizados 4](#_Toc198593606)

[2.1. Imágenes Docker: 4](#_Toc198593607)

[2.2. Docker Compose: 8](#_Toc198593608)

[3. Configuración para Kubernetes 10](#_Toc198593609)

[3.1. Archivos YAML: 10](#_Toc198593610)

[4. Microservicios. 16](#_Toc198593611)

[4.1. Manifiestos completos: 17](#_Toc198593612)

[5. Bases de Datos 18](#_Toc198593613)

[5.1. Scripts de Inicialización: 18](#_Toc198593614)

[5.2. Volúmenes: 24](#_Toc198593615)

[6. Documentación 24](#_Toc198593616)

[6.1. Manual de Despliegue: 24](#_Toc198593617)

[6.2. Diagrama de Arquitectura: 26](#_Toc198593618)

[7. Resultados 26](#_Toc198593619)

[7.1. Validación del Despliegue: 26](#_Toc198593620)

[7.2. Registro de Imágenes: 32](#_Toc198593621)

# Proyecto Curso Docker y Kubernetes

## Entregables del Proyecto Final: Gestión de Eventos y Participantes

Los entregables se organizan en categorías que abarcan desde el código fuente hasta la documentación y los artefactos de despliegue.

Se creó el siguiente repositorio Git para todos los entregables del presente proyecto: <https://github.com/jonathan0284/Curso_Docker_K8S.git>

### Código Fuente

### Repositorios de Microservicios:

* Código fuente de los dos Micro servicios: gestión de eventos y gestión de inscripciones (participantes).

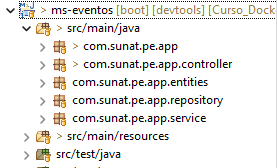
ms-eventos: <https://github.com/jonathan0284/Curso_Docker_K8S/tree/main/Microservicios/ms-eventos>

ms-participantes:

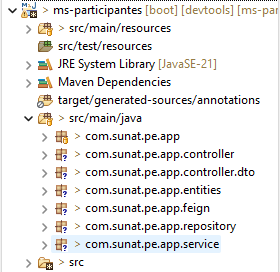
<https://github.com/jonathan0284/Curso_Docker_K8S/tree/main/Microservicios/ms-participantes>

* Estructura del proyecto conforme a las mejores prácticas (separación en capas: controlador, servicio, repositorio, entidades, etc.).

**ms-eventos**



**ms-participantes**



* Uso de control de versiones (Git) con un historial claro de commits: <https://github.com/jonathan0284/Curso_Docker_K8S/activity>

### Artefactos Contenerizados

### Imágenes Docker:

* Imágenes Docker de ambos microservicios, construidas y publicadas en un repositorio como Docker Hub.

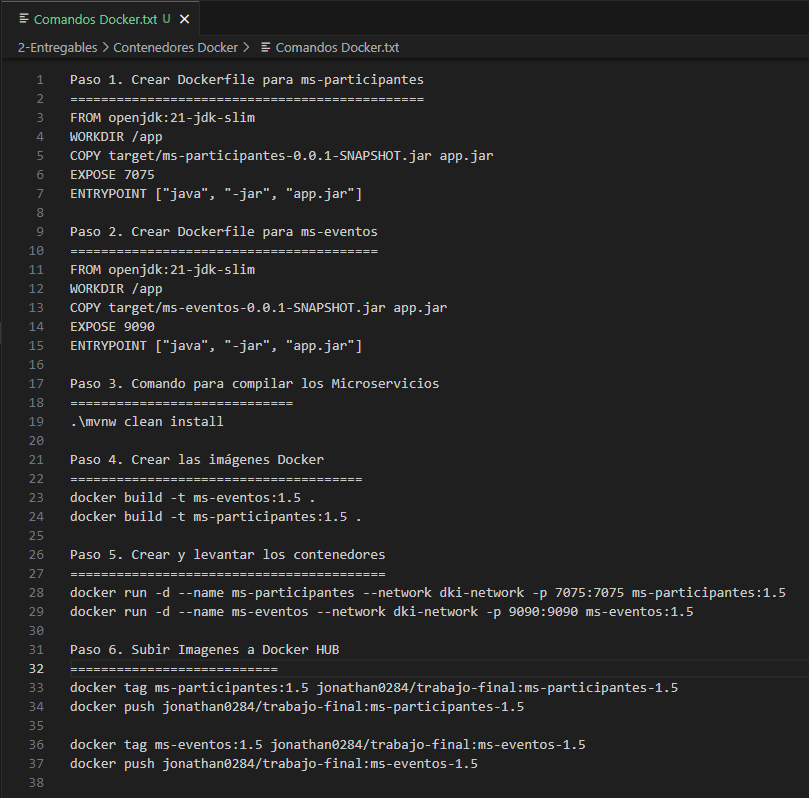


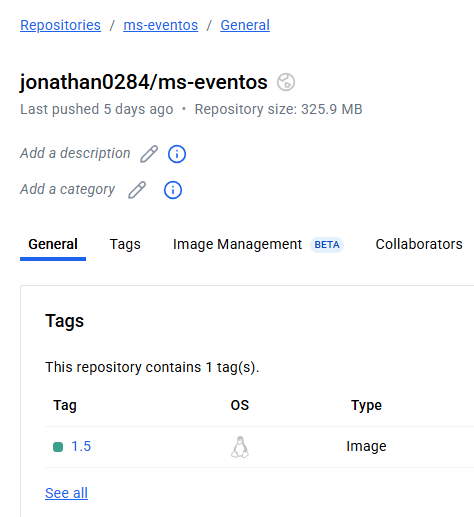
**ms-eventos**

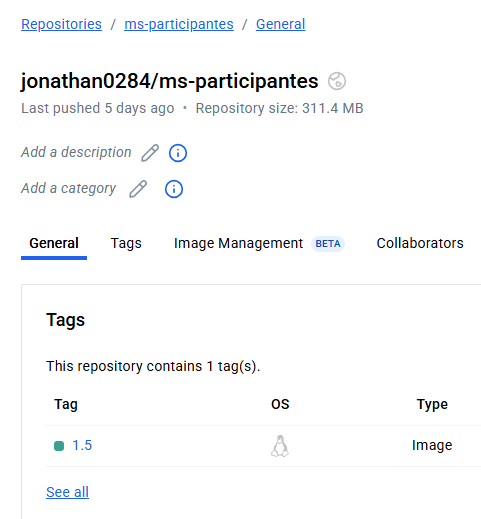
<https://hub.docker.com/repository/docker/jonathan0284/ms-eventos>

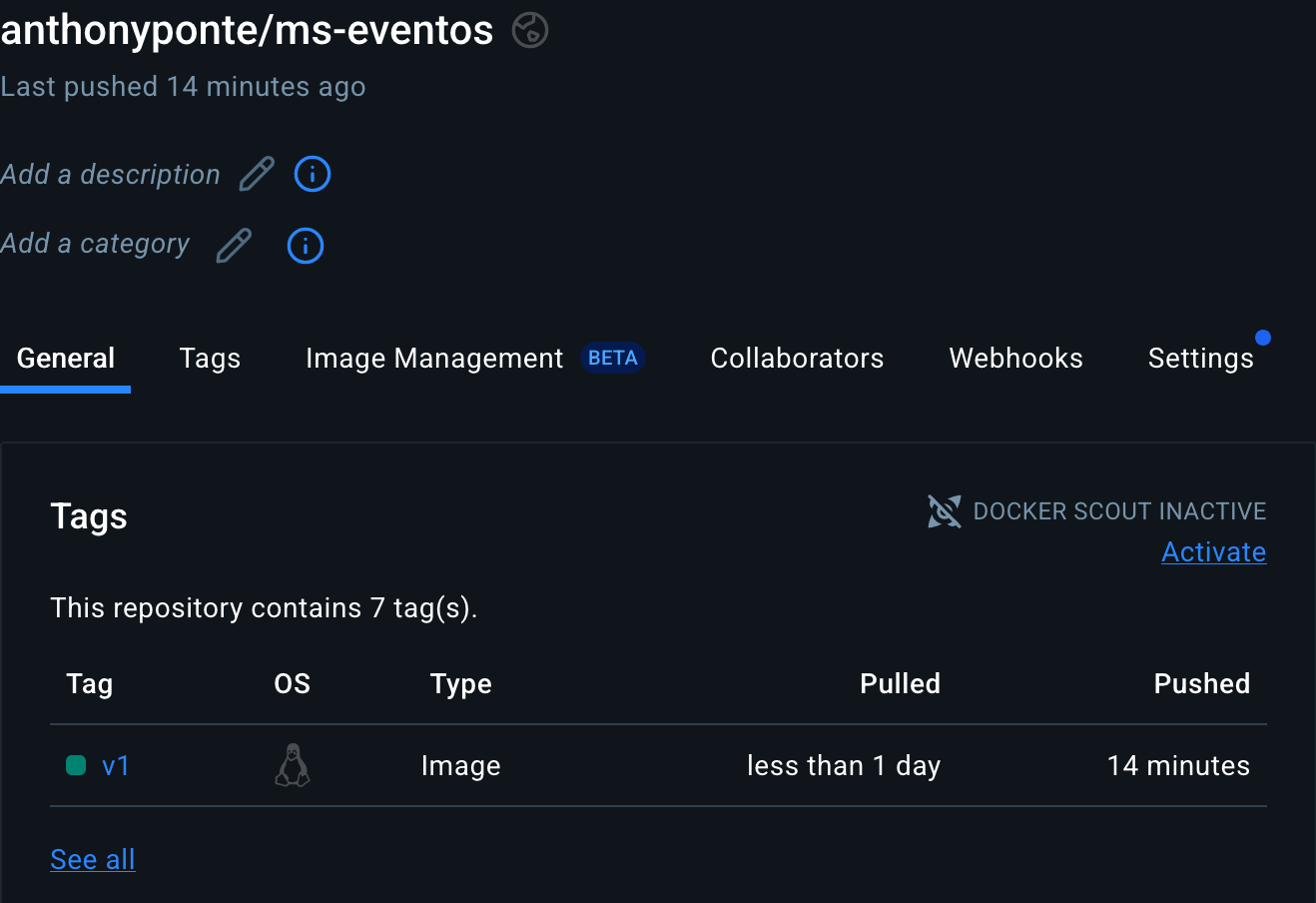
**ms-participantes**

<https://hub.docker.com/repository/docker/jonathan0284/ms-participantes/general>





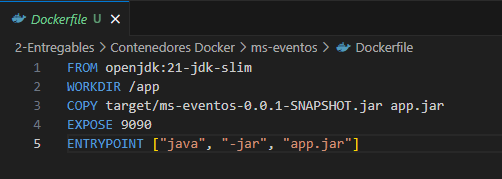




* Imágenes optimizadas según las mejores prácticas (Dockerfile eficiente).

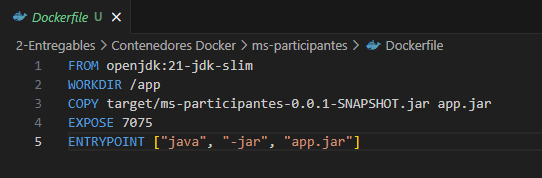
**ms-eventos**





**ms-participantes**

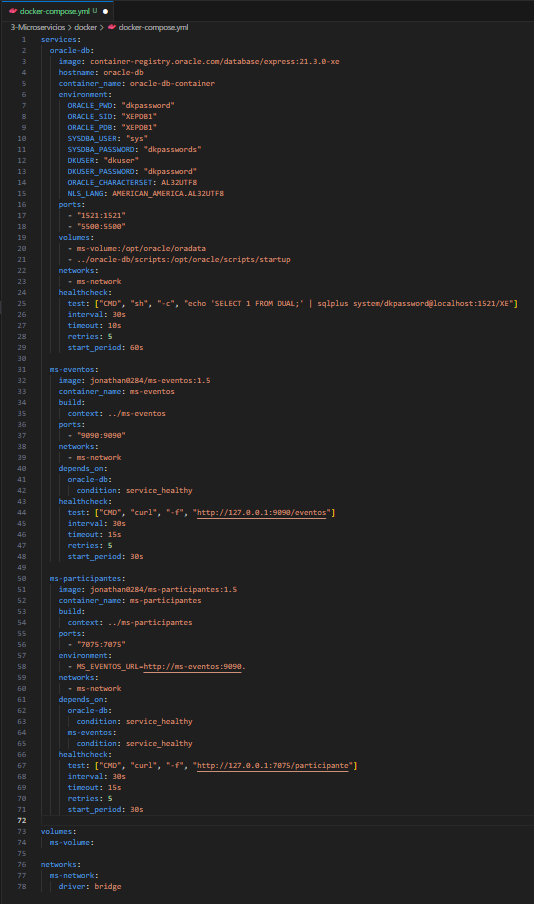




### Docker Compose:

* Archivo docker-compose.yml para levantar ambos microservicios junto con sus bases de datos en un entorno de desarrollo local.





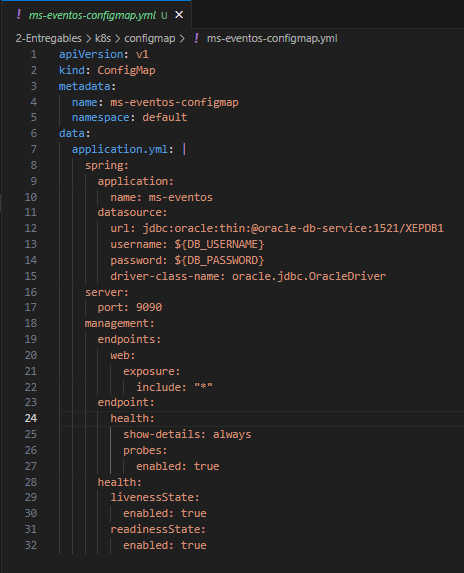
### Configuración para Kubernetes

### Archivos YAML:

* **Deployments:** Configuración de pods y réplicas para ambos microservicios.

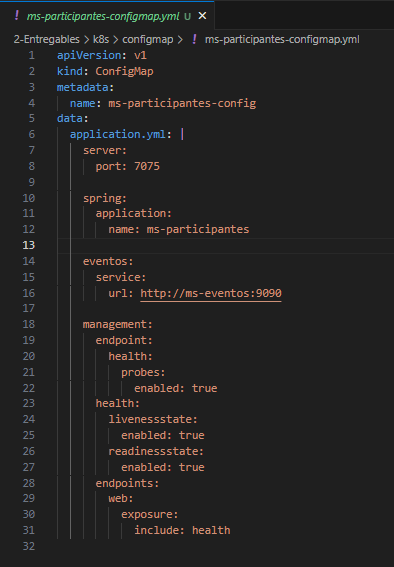
**ms-eventos-configmap.yml**

****



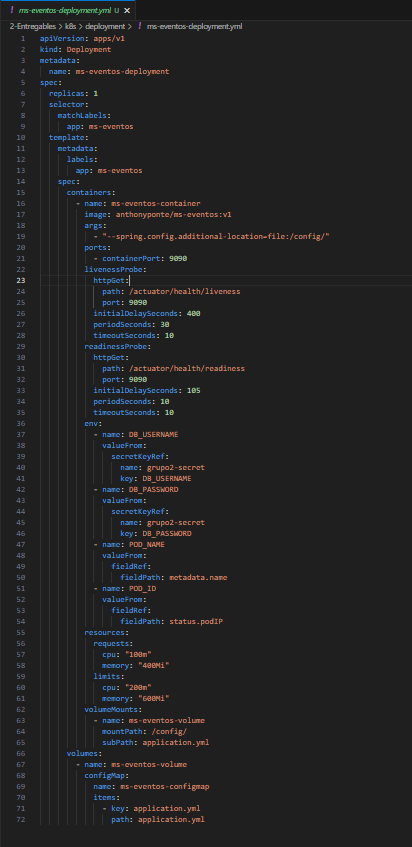
**ms-participantes-configmap.yaml**

****



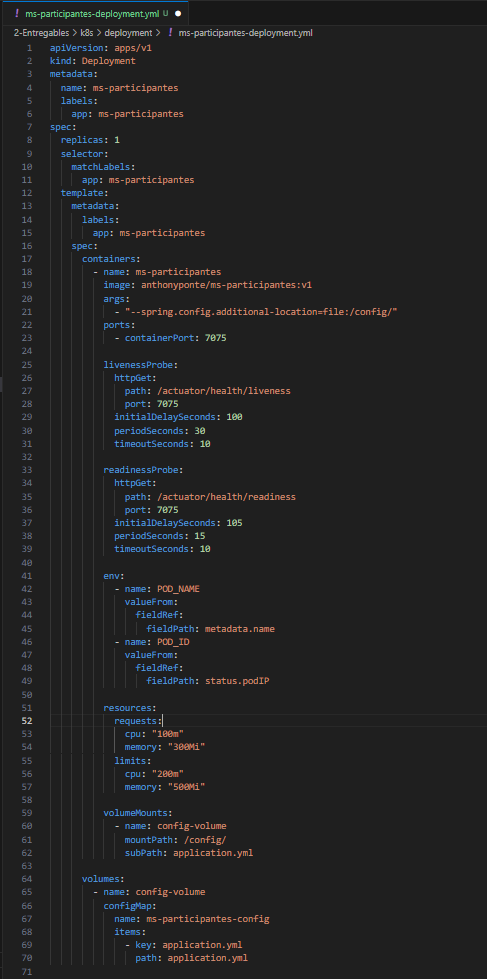
**ms-eventos-deployment.yml**

****



**ms-participantes-deployment.yml**

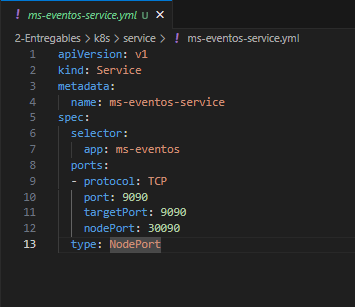
****



* **Services:** Definición de servicios ClusterIP o NodePort para los

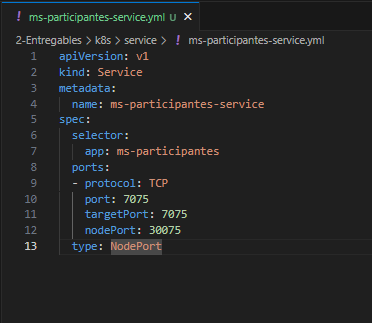
**ms-eventos-service.yml**

****



**ms-participantes-service.yml**

****

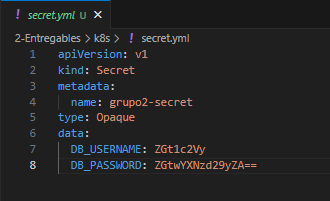


### Microservicios.

* **Ingress:** Configuración de rutas basadas en contexto para exponer las APIs externamente (Opcional, este tema se cubrió en en nivel Básico)
* **ConfigMaps y Secrets:** Manejo de variables de entorno y datos sensibles.

**secret.yml**

****

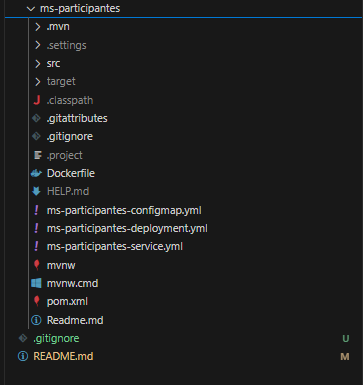


* **Probes:** Liveness y readiness probes configuradas.

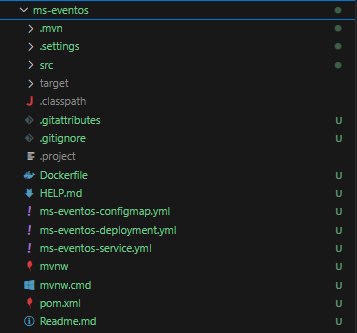
### Manifiestos completos:

* Conjunto de manifiestos YAML organizados y documentados en carpetas para cada microservicio.

**ms-participantes**



**ms-eventos**



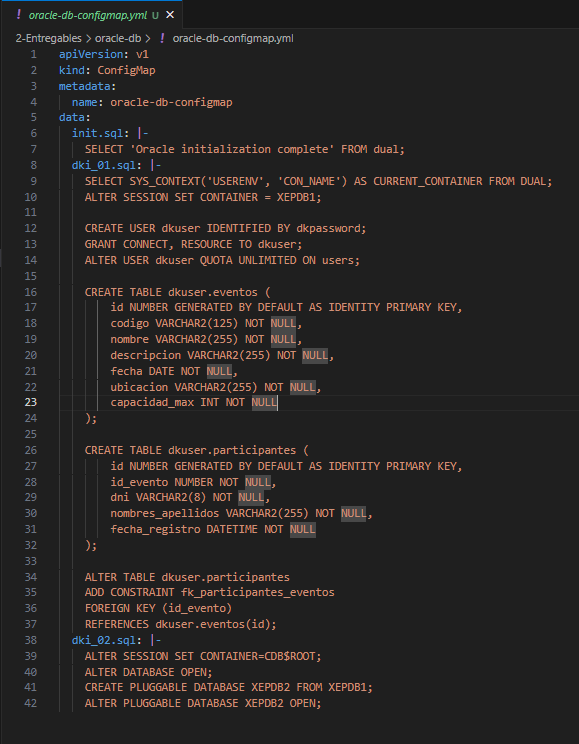
### Bases de Datos

### Scripts de Inicialización:

* Scripts SQL para inicializar las bases de datos con tablas necesarias y datos de prueba.

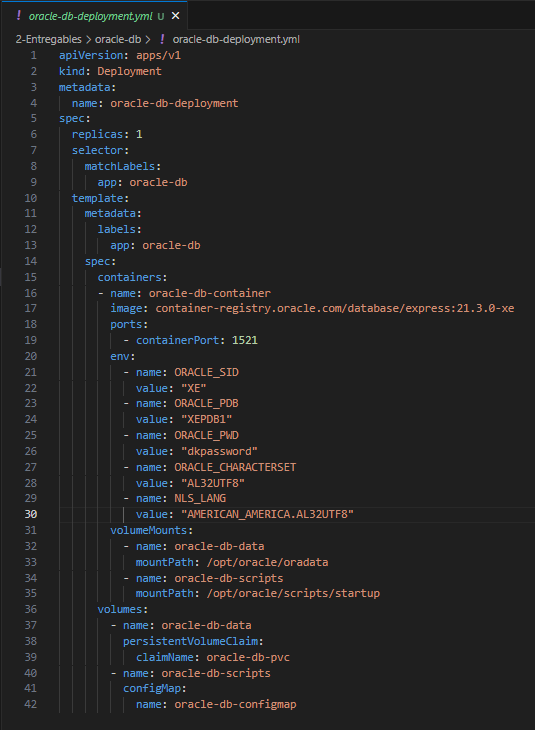
**oracle-db-configmap.yml**

****



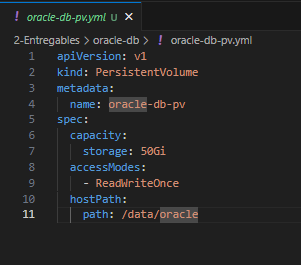
**oracle-db-deployment.yml**

****



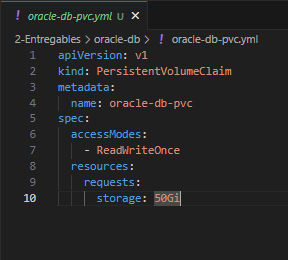
**oracle-db-pv.yml**

****



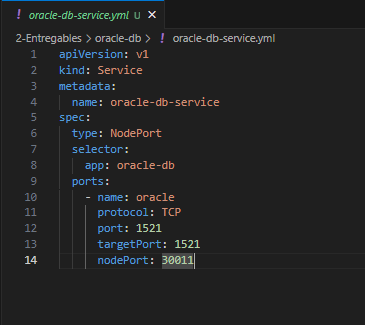
**oracle-db-pvc.yml**

****



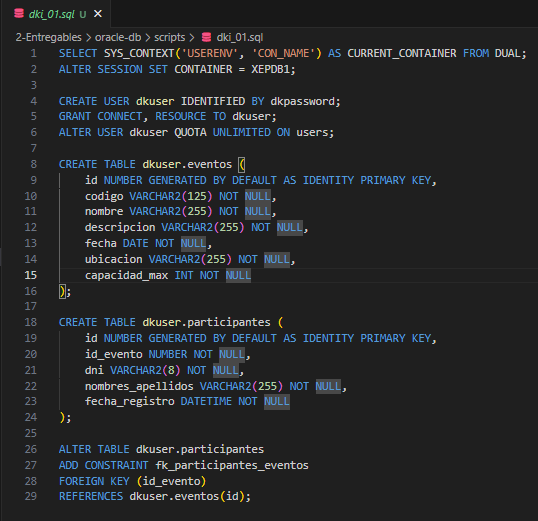
**oracle-db-service.yml**

****



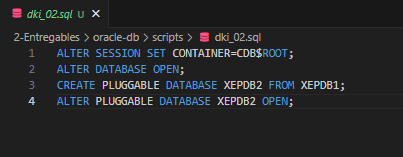
**dki\_01.sql**

****



**dki\_02.sql**

****



### Volúmenes:

* Configuración de volúmenes en Kubernetes para garantizar la persistencia de los datos.

### Documentación

### Manual de Despliegue:

* Pasos detallados para:

▪ Contenerizar y construir las imágenes:

*Verificar la existencia de la imagen dki-volume:*

*docker volume ls*

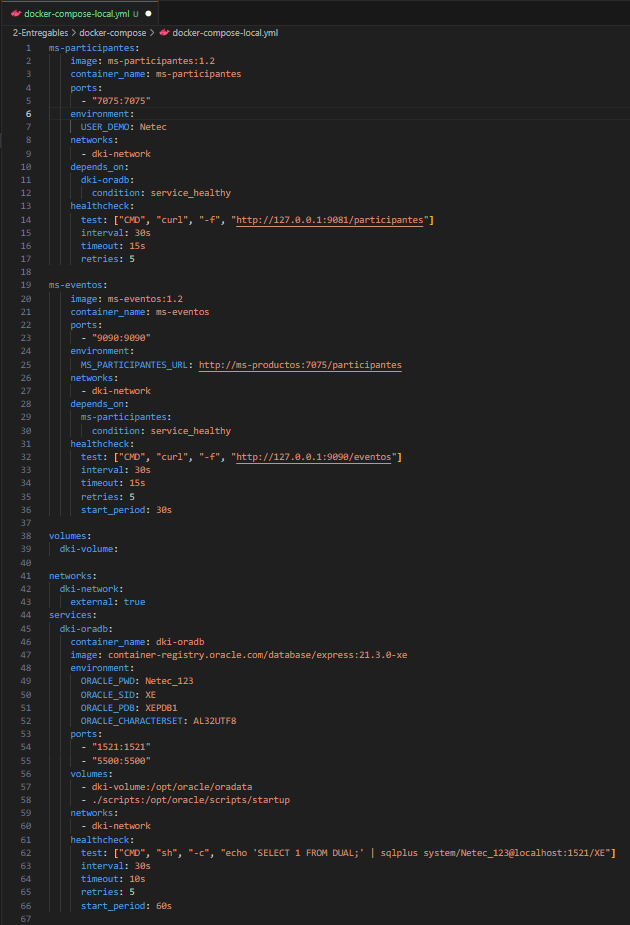
*Creación de la imagen dki-volume:*

*docker volume create dki-volume*

▪ Desplegar en Kubernetes utilizando los manifiestos YAML.

▪ Configurar Docker Compose para desarrollo local.

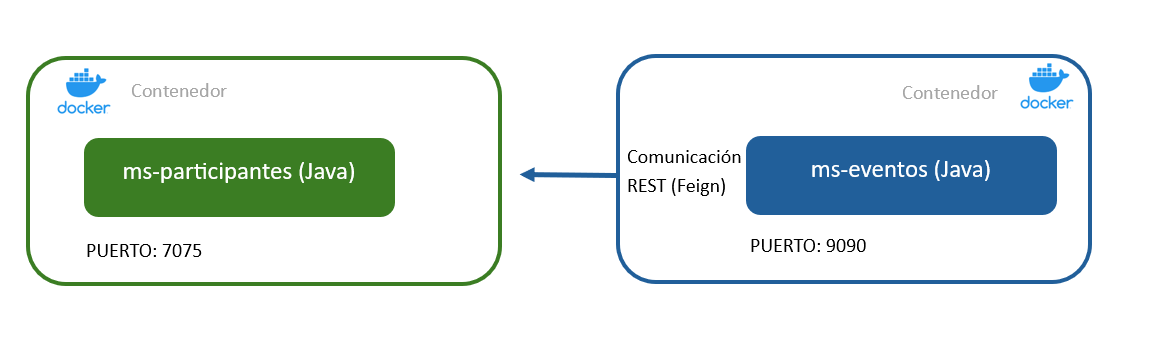




### Diagrama de Arquitectura:

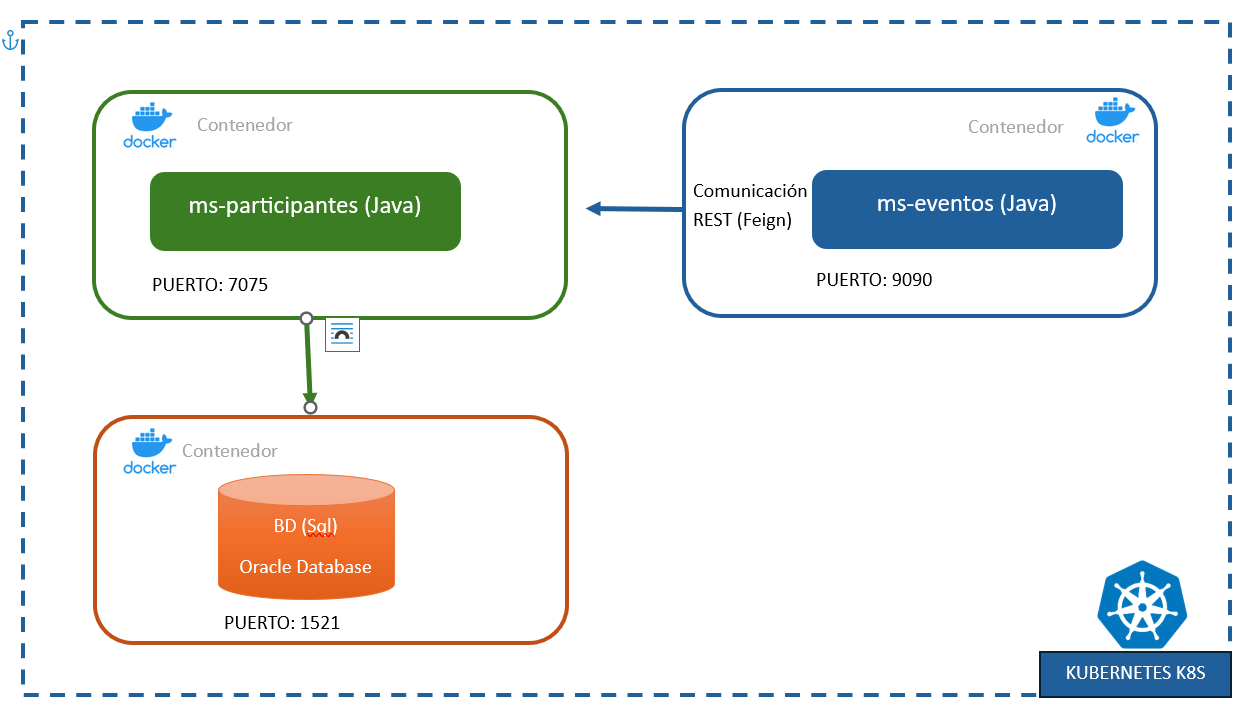
* Representación gráfica de la solución, incluyendo:

▪ Estructura de microservicios:



▪ Comunicación entre ellos.

▪ Despliegue en Kubernetes:



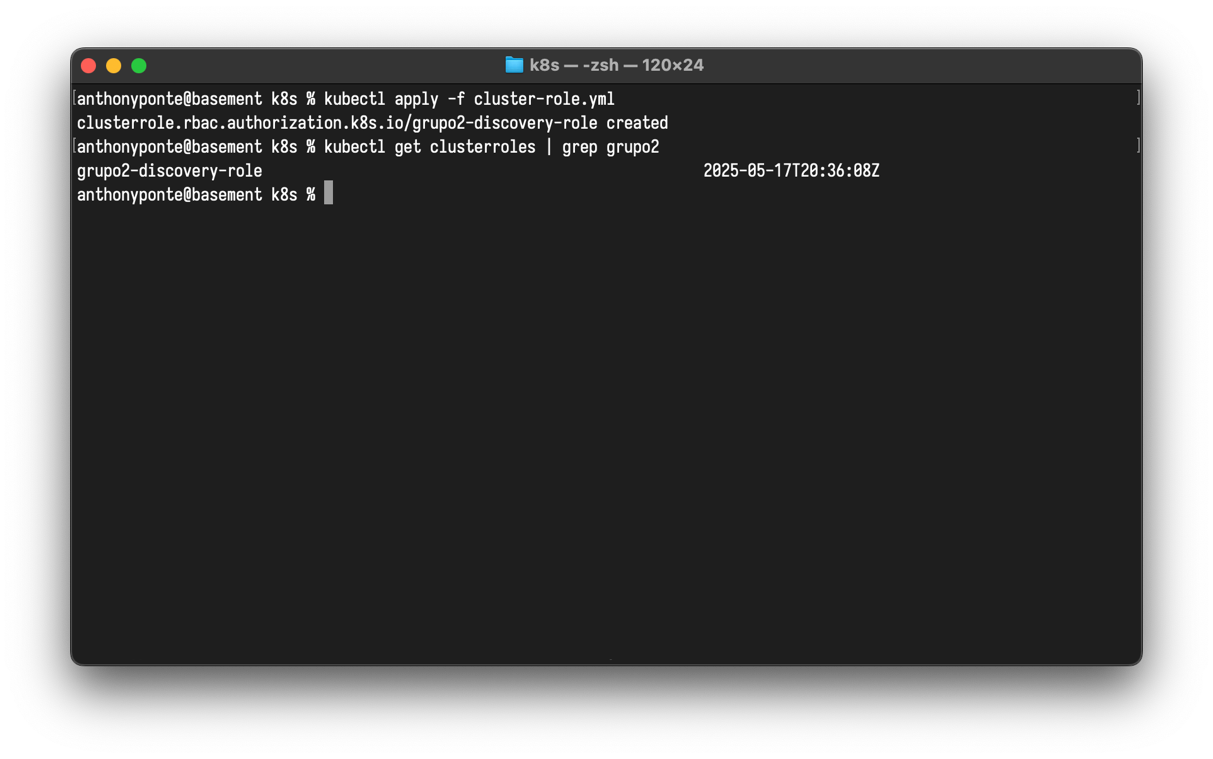
### Resultados

### Validación del Despliegue:

* Evidencia del despliegue exitoso en Kubernetes (por ejemplo, capturas de pantalla o logs que muestren los pods corriendo).

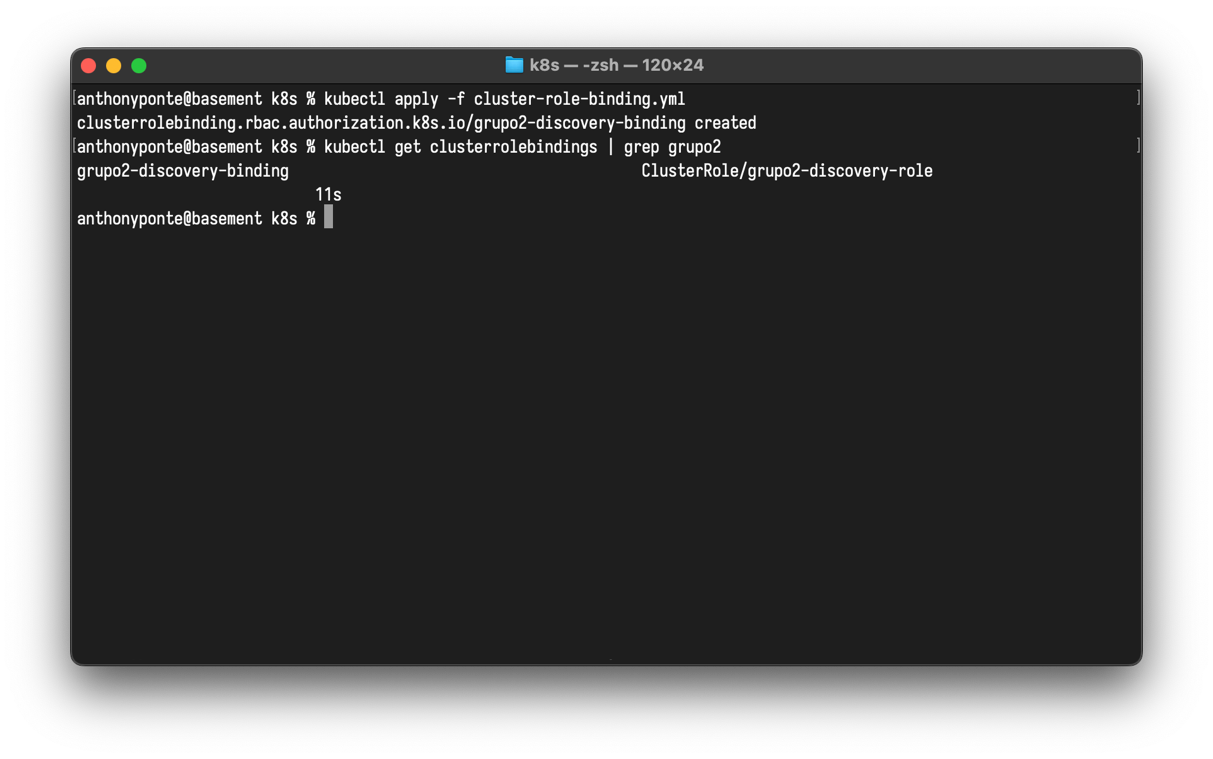
kubectl apply -f cluster-role.yml

kubectl get clusterroles | grep grupo2



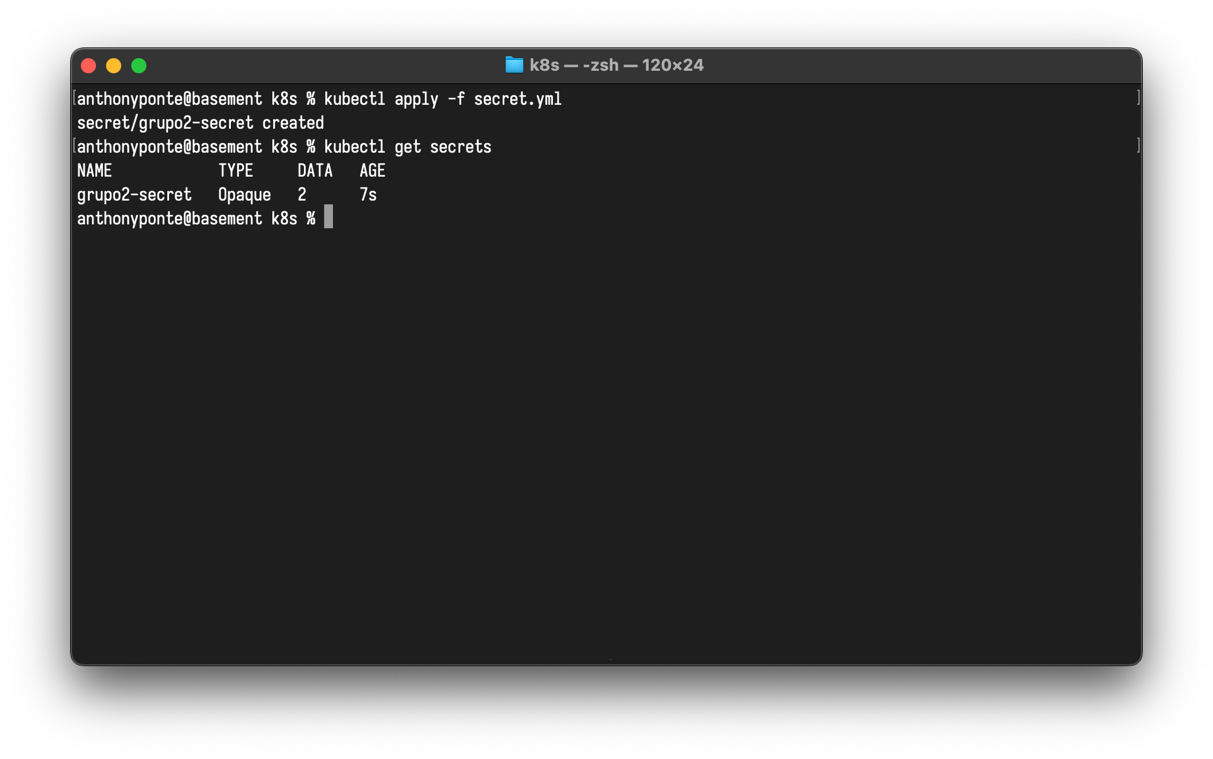
kubectl apply -f cluster-role-binding.yml

kubectl get clusterrolebindings | grep grupo2



kubectl apply -f secret.yml

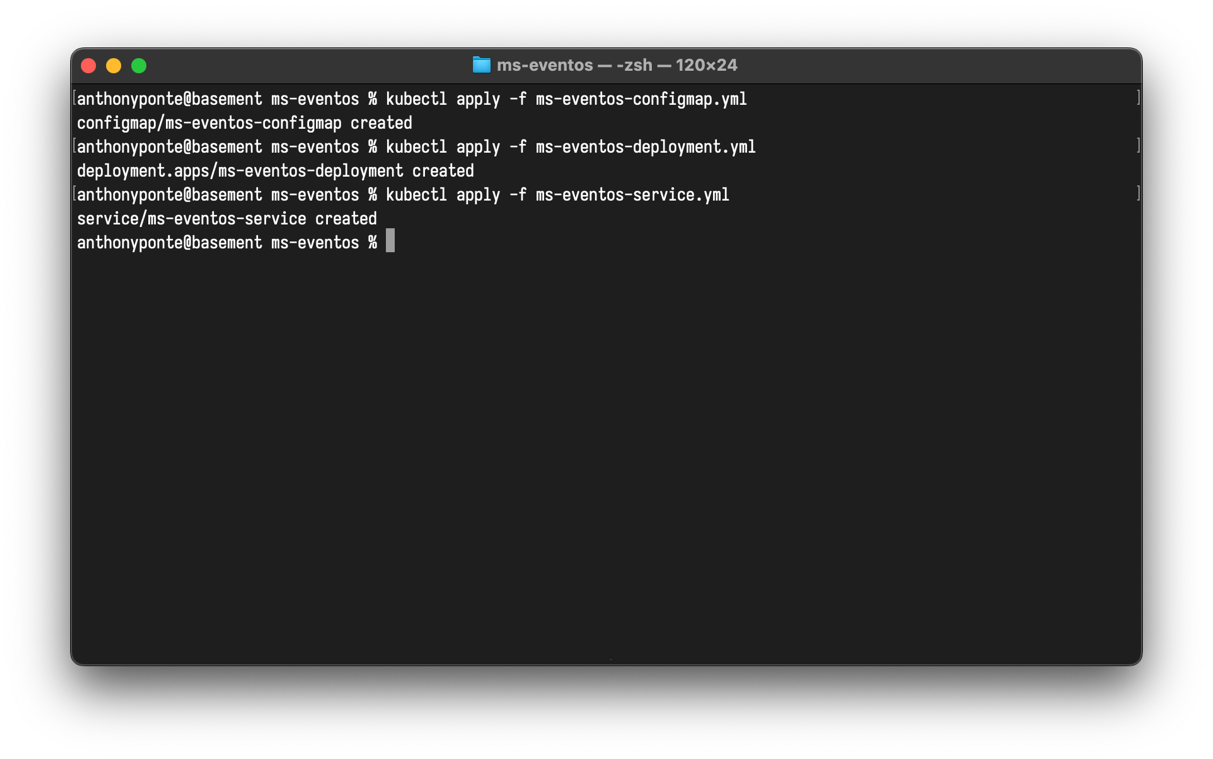
kubectl get secrets



kubectl apply -f ms-eventos-configmap.yml

kubectl apply -f ms-eventos-deployment.yml

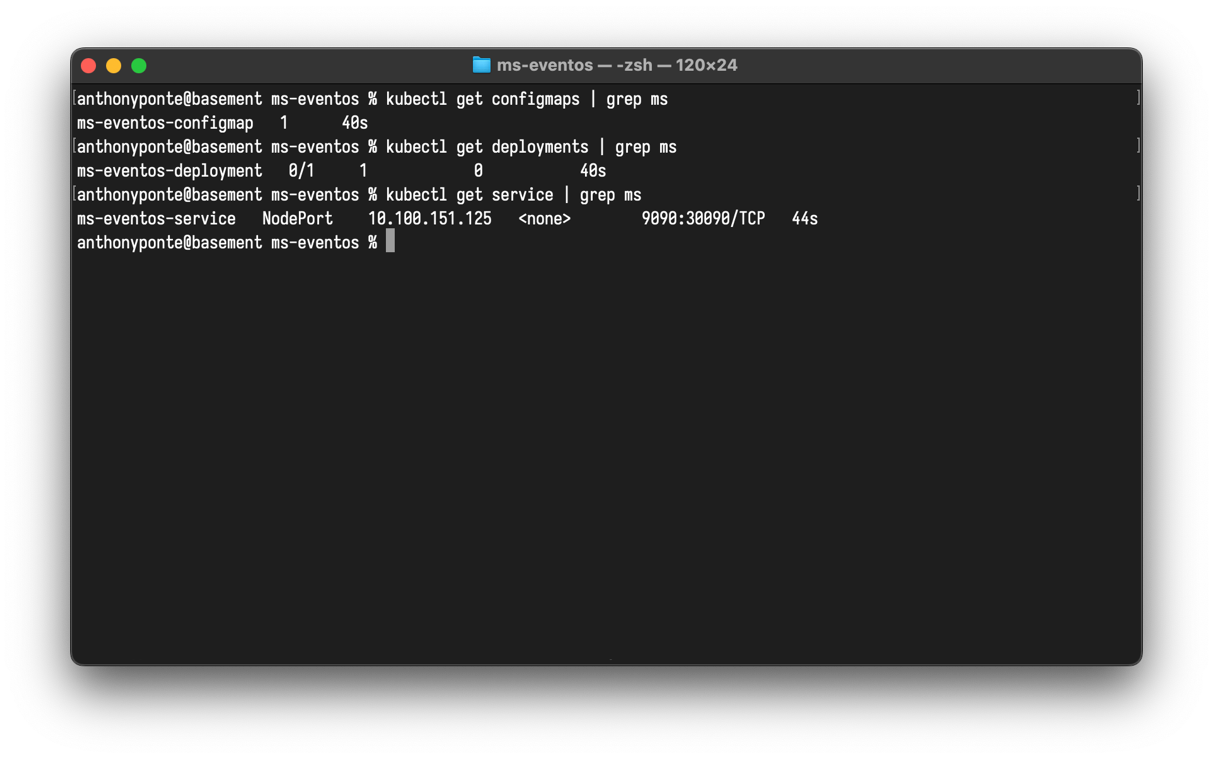
kubectl apply -f ms-eventos-service.yml



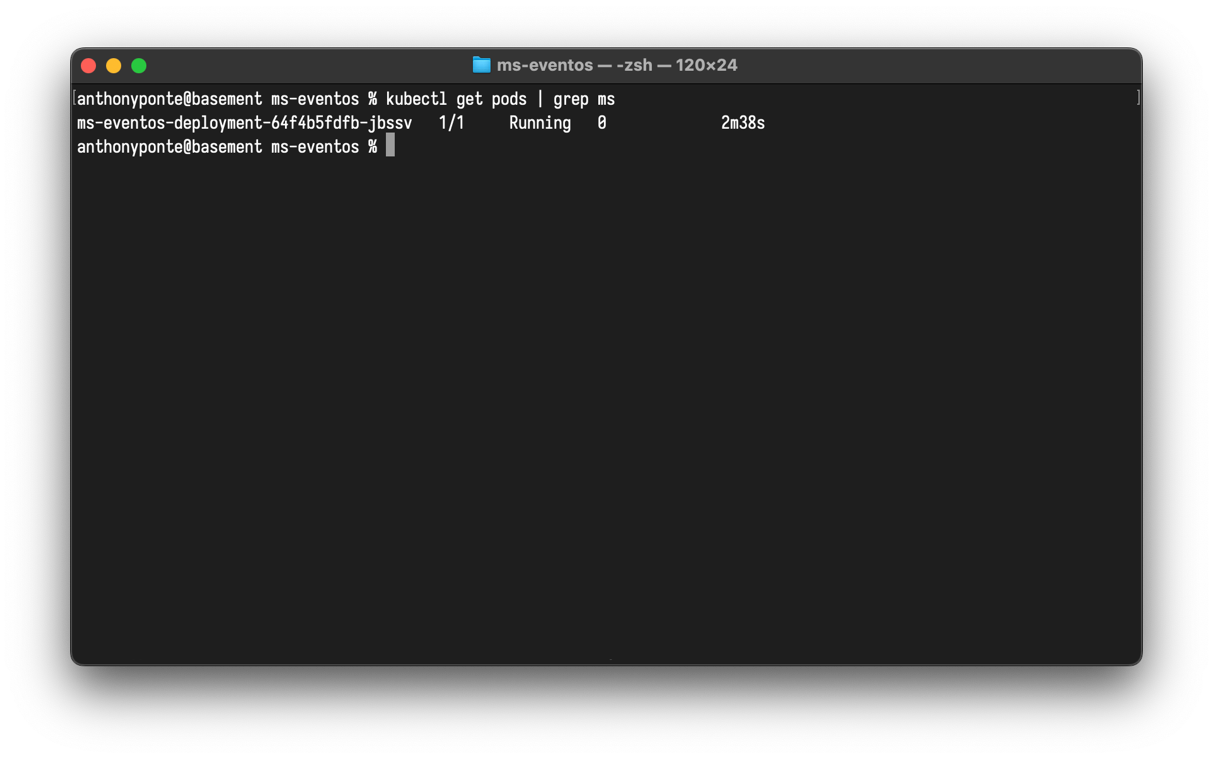
kubectl get configmaps | grep ms

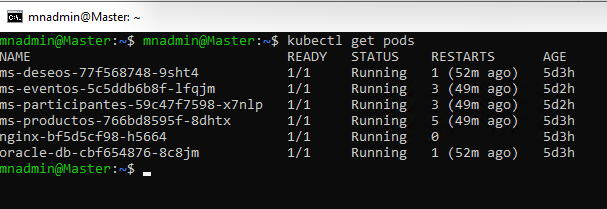
kubectl get deployments | grep ms

kubectl get service | grep ms

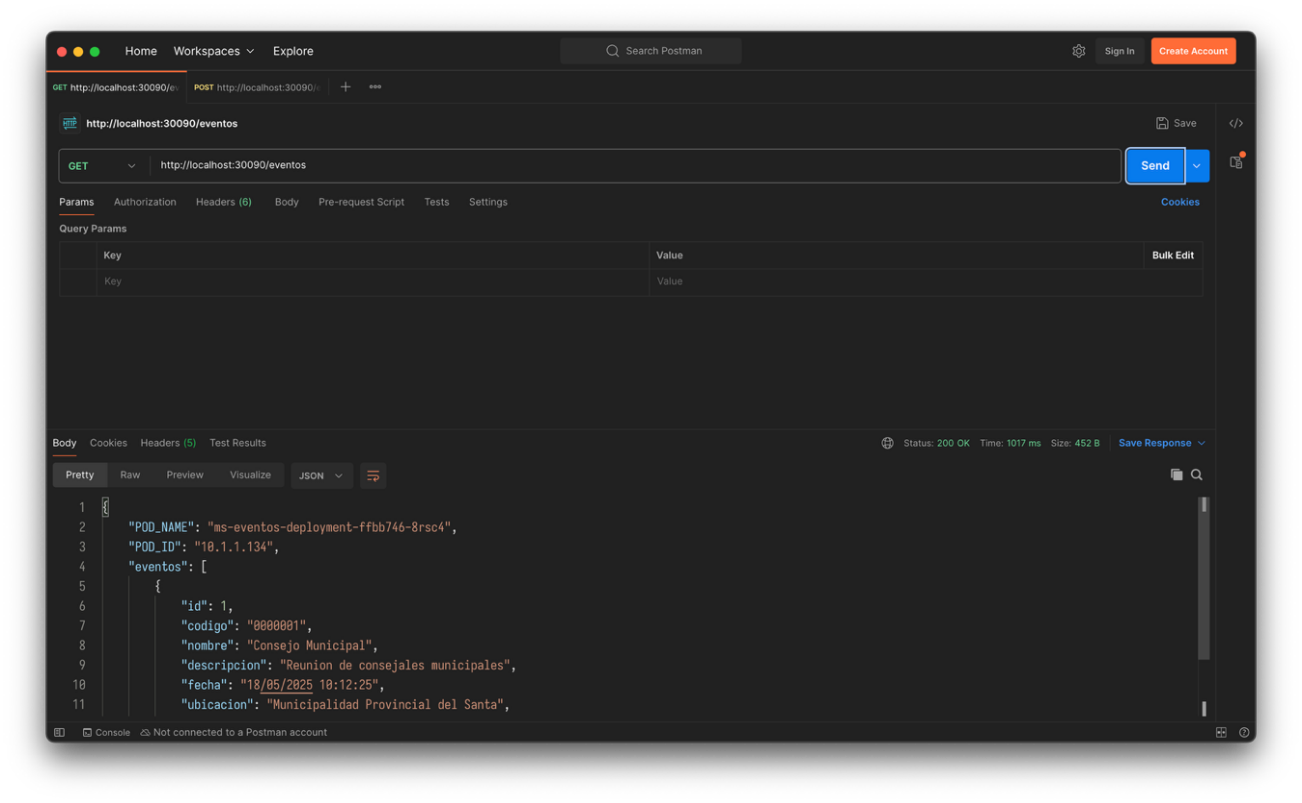


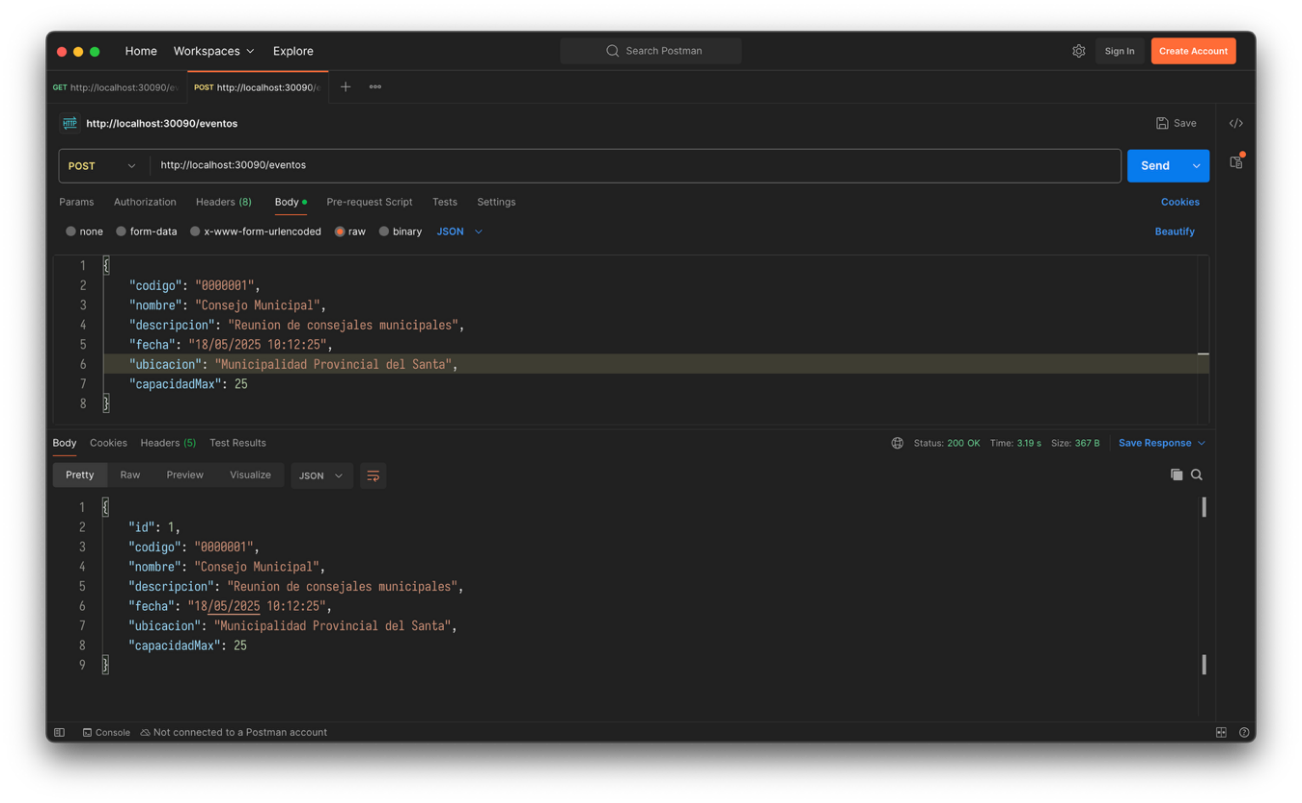
kubectl get pods | grep ms

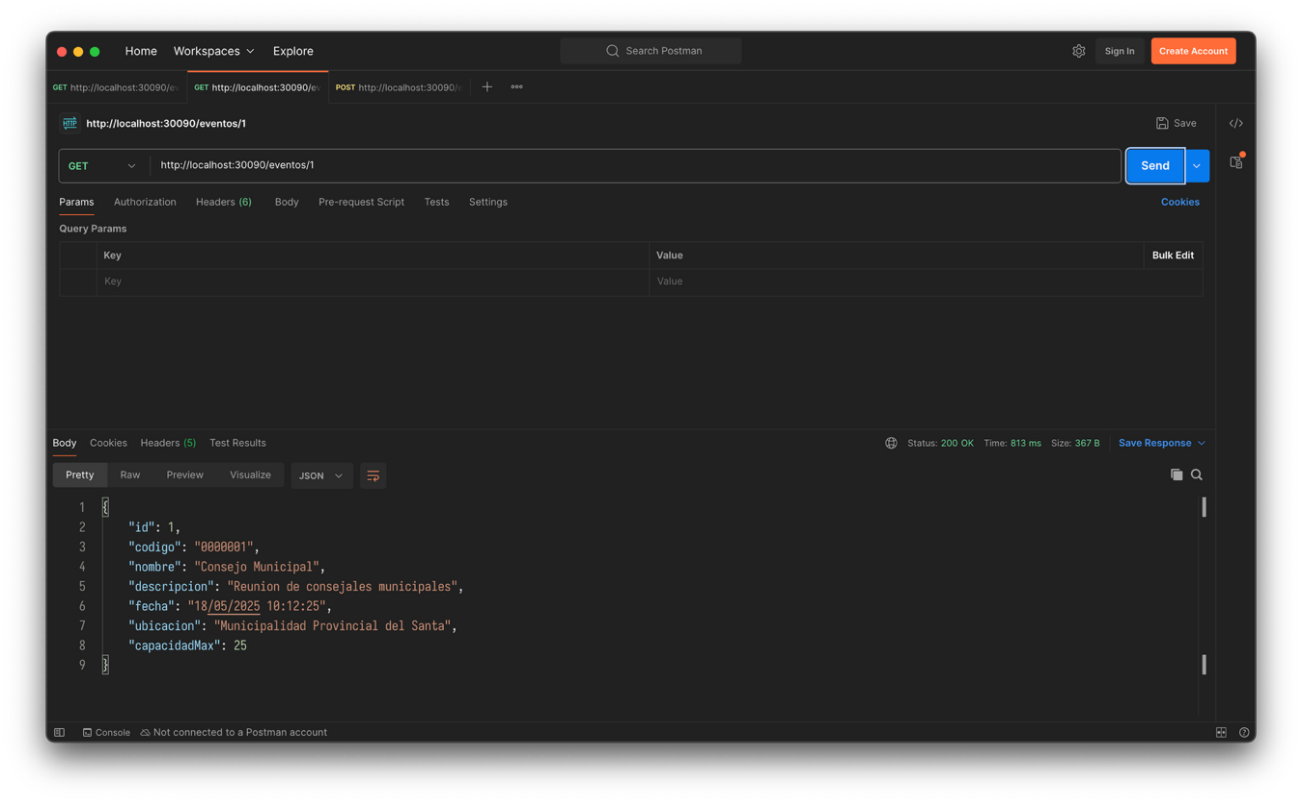


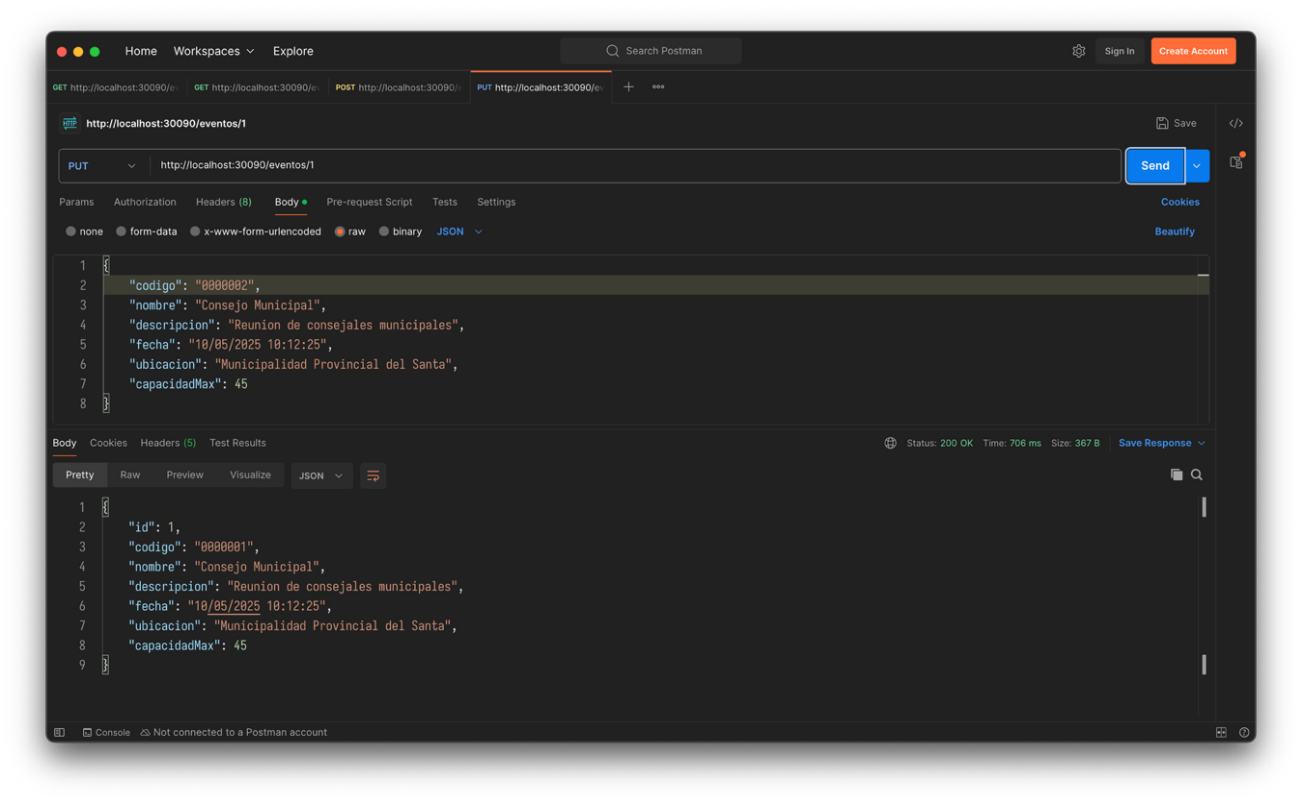


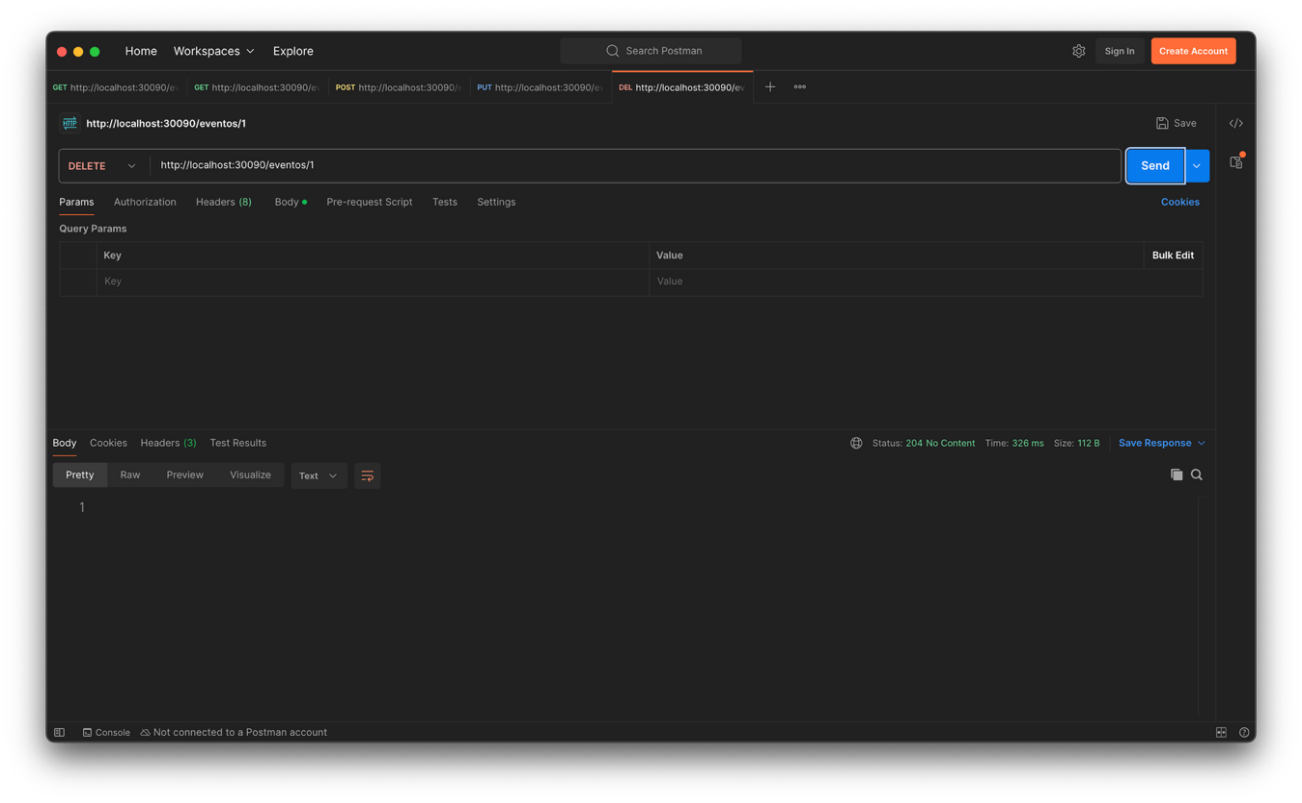
* Pruebas realizadas en los servicios mediante Postman o CURL.











### Registro de Imágenes:

* URL de las imágenes Docker publicadas (Docker Hub).

**ms-eventos:**

<https://hub.docker.com/repository/docker/jonathan0284/ms-eventos/general>

**ms-participantes:**

<https://hub.docker.com/repository/docker/jonathan0284/ms-participantes/general>