

## 1. kb-eventos (Base de datos de eventos)

Archivos en la carpeta kb-eventos:

- configmap-db-eventos.yaml
- pv-db-eventos.yaml
- pvc-db-eventos.yaml
- deployment-db-eventos.yaml
- service-db-eventos.yaml

Comandos:

```
cd kb-namespace
```

```
kubectl apply -f namespace-gestion-ep-dev.yaml
```

```
cd kb-eventos
```

```
# Crear el ConfigMap para la base de datos de eventos
```

```
kubectl apply -f configmap-db-eventos.yaml -n gestion-ep-dev
```

```
# Crear el Persistent Volume (PV) para la base de datos de eventos
```

```
kubectl apply -f pv-db-eventos.yaml -n gestion-ep-dev
```

```
# Crear el Persistent Volume Claim (PVC) para la base de datos de eventos
```

```
kubectl apply -f pvc-db-eventos.yaml -n gestion-ep-dev
```

```
# Desplegar la base de datos de eventos
```

```
kubectl apply -f deployment-db-eventos.yaml -n gestion-ep-dev
```

```
# Crear el servicio para la base de datos de eventos
```

```
kubectl apply -f service-db-eventos.yaml -n gestion-ep-dev
```

---

## 2. kb-participantes (Base de datos de participantes)

Archivos en la carpeta kb-participantes:

- configmap-db-participantes.yaml
- pv-db-participantes.yaml
- pvc-db-participantes.yaml
- deployment-db-participantes.yaml
- service-db-participantes.yaml

Comandos:

```
cd ../kb-participantes
```

```
# Crear el ConfigMap para la base de datos de participantes
```

```
kubectl apply -f configmap-db-participantes.yaml -n gestion-ep-dev
```

```
# Crear el Persistent Volume (PV) para la base de datos de  
participantes
```

```
kubectl apply -f pv-db-participantes.yaml -n gestion-ep-dev
```

```
# Crear el Persistent Volume Claim (PVC) para la base de datos de  
participantes
```

```
kubectl apply -f pvc-db-participantes.yaml -n gestion-ep-dev
```

```
# Desplegar la base de datos de participantes
```

```
kubectl apply -f deployment-db-participantes.yaml -n gestion-ep-dev
```

```
# Crear el servicio para la base de datos de participantes
```

```
kubectl apply -f service-db-participantes.yaml -n gestion-ep-dev
```

---

### 3. kb-microservicios (Microservicios de eventos y participantes)

Archivos en la carpeta kb-microservicios:

- ms-eventos-deployment.yaml
- ms-participantes-deployment.yaml
- ms-eventos-service.yaml
- ms-participantes-service.yaml
- ms-eventos-configmap.yaml
- ms-participantes-configmap.yaml

Comandos:

```
cd ../kb-microservicios
```

```
# Paso 1: Crear los ConfigMaps
```

```
kubectl apply -f configmap-ms-eventos.yaml -n gestion-ep-dev
```

```
kubectl apply -f configmap-ms-participantes.yaml -n gestion-ep-dev
```

```
# Paso 2: Crear el Secret para la base de datos
```

```
kubectl apply -f secret-db.yaml -n gestion-ep-dev
```

```
# Paso 3: Desplegar los microservicios
```

```
kubectl apply -f deployment-ms-eventos.yaml -n gestion-ep-dev
```

```
kubectl apply -f deployment-ms-participantes.yaml -n gestion-ep-dev
```

```
# Paso 4: Crear los servicios para los microservicios
```

```
kubectl apply -f service-ms-eventos.yaml -n gestion-ep-dev
```

```
kubectl apply -f service-ms-participantes.yaml -n gestion-ep-dev
```

Verificar el estado del despliegue:

Después de ejecutar estos comandos, puedes verificar si todo se desplegó correctamente:

# Verificar los deployments

kubectl get deployments -n gestion-ep-dev

# Verificar los pods para cada servicio

kubectl get pods -n gestion-ep-dev

# Verificar los servicios

kubectl get services -n gestion-ep-dev

Orden correcto de ejecución:

1. kb-eventos: Desplegar la base de datos de eventos primero.
2. kb-participantes: Después, desplegar la base de datos de participantes.
3. kb-microservicios: Finalmente, desplegar los microservicios de eventos y participantes.