

# 🧩 Del commit al despliegue sin drama

Autor: Rubén López

Misión: La Huella — Etapa 7

## 1. Punto de partida

Para este ejercicio, partimos de una infraestructura ya desplegada previamente en el clúster de Minikube (lahuella). En etapas anteriores se instalaron y configuraron:

- Localstack (vía Helm), simulando servicios de AWS como S3, DynamoDB y SQS.
- Nginx Ingress Controller, exponiendo los servicios bajo el dominio local midominio.local.
- Terraform, que creó los recursos necesarios dentro de Localstack y almacenó su estado en un bucket

Por tanto, el objetivo de este ejercicio no ha sido desplegar la infraestructura, sino hacer que la aplicación cobre vida sobre ella, cerrando el ciclo completo de desarrollo y despliegue continuo.

### 2. Preparación de la aplicación

El primer paso fue ajustar la configuración de la aplicación para que utilizara los endpoints del entorno ya existente:

Se modificaron las variables de entorno para apuntar al endpoint de Localstack:

```
AWS_ENDPOINT_URL=http://midominio.local
```

- En next.config.js se añadió el dominio a los allowedOrigins.
- En los YAML de despliegue de Kubernetes se configuró también esta URL como variable de entorno para los pods.

De este modo, la aplicación puede comunicarse correctamente con los servicios simulados (S3, DynamoDB y SQS) de Localstack expuestos en http://midominio.local.

## 3. Definición de los manifiestos de Kubernetes

A continuación, se crearon los manifiestos YAML necesarios para el despliegue de la aplicación dentro del clúster:

**Archivo** 

Descripción

Archivo	Descripción
deployment- app.yaml	Define el pod que ejecuta la aplicación (imagen Docker, variables de entorno y puertos).
service-app.yaml	Expone la aplicación dentro del clúster en el puerto 80.
ingress-app.yaml	Publica la aplicación a través del dominio midominio.local/app.

**Importante:** Ya existía un Ingress que utilizaba midominio.local/ para Localstack, por lo que se decidió servir la aplicación bajo la ruta /app, evitando conflicto con el path raíz /.

Así, tras aplicar los manifiestos:

- Localstack es accesible en http://midominio.local/
- La aplicación en http://midominio.local/app

# 4. Pipeline CI/CD con GitHub Actions

Para automatizar el despliegue, se configuró un workflow en GitHub Actions (.github/workflows/deploy.yaml) con las siguientes etapas:

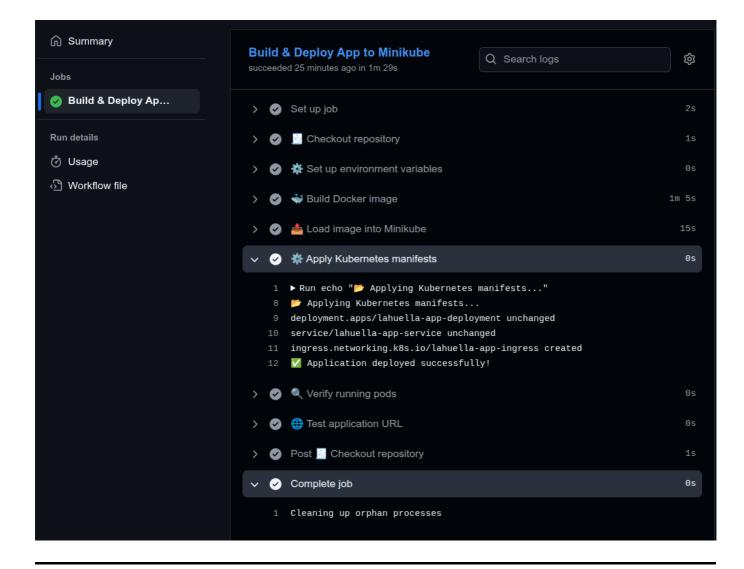
- 1. Checkout del repositorio.
- 2. Verificación del estado del clúster Minikube.
  - Se asegura que el perfil lahuella está corriendo (minikube -p lahuella status).
- 3. Construcción de la imagen Docker a partir del Dockerfile incluido en el repositorio.
- 4. Carga de la imagen en Minikube:

```
minikube -p lahuella image load lahuella-app:latest
```

5. Despliegue de los manifiestos Kubernetes:

```
kubectl apply -f infra/k8s/
```

6. Verificación del estado del pod y del Ingress.



## 5. Registro y ejecución del runner local

Para ejecutar el pipeline, se registró un runner self-hosted en el repositorio de GitHub

(RubenLopSol/eu-devops-7-la-huella):

```
./config.sh --url https://github.com/RubenLopSol/eu-devops-7-la-huella --
token <TOKEN>
./run.sh
```

El runner se asoció al repositorio y quedó en estado **Online (Idle)**, esperando jobs. De esta forma, cada git push a la rama main dispara automáticamente el despliegue de la aplicación sobre el clúster local.

### 6. Inserción de datos en la base de datos simulada

Antes de ejecutar el workflow, se pobló DynamoDB con datos de ejemplo mediante el script:

```
./scripts/init.sh
```

```
terraform git:(main) x awslocal --endpoint-url=http://midominio.local dynamodb scan --table-name la-huella-products | jq '.Items | length'

terraform git:(main) x [
```

Este script inserta productos y comentarios en las tablas creadas por Terraform dentro de Localstack, dejando la infraestructura lista para ser consumida por la aplicación.

## 7. Despliegue final y validación

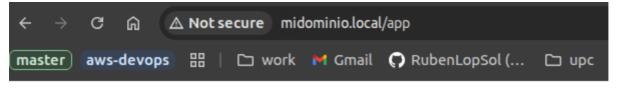
Tras ejecutar el workflow, los pasos fueron completados con éxito:

- Imagen Docker construida correctamente.
- Carga de la imagen en Minikube (lahuella).
- Creación del Deployment, Service e Ingress.
- Pod lahuella-app-deployment en estado Running.
- Ingress visible en:

```
get ingress -A
NAMESPACE
              NAME
                                      CLASS
                                               HOSTS
                                                                  ADDRESS
                                                                                   PORTS
                                                                                           AGE
              lahuella-app-ingress
default
                                               midominio.local
                                                                  192.168.49.2
                                                                                   80
                                                                                           39m
                                      nginx
              localstack-ingress
                                      nginx
                                               midominio.local
                                                                  192.168.49.2
                                                                                   80
                                                                                           4h53m
```

#### Finalmente, se validó el acceso:

- http://midominio.local/ → Localstack Dashboard
- http://midominio.local/app → Aplicación desplegada y conectada a Localstack



LH

## La Huella

análisis de sentimiento

<u>DashboardProductosAnalytics</u>



Monitorea y analiza los comentarios de productos de calzado

Pipeline Básico

**Total Productos** 

5

**Total Comentarios** 

648

Rating Promedio

4.1/5.0

Satisfacción

65%

**T** • • •

### 8. Conclusión

Este ejercicio cierra el ciclo completo "Del commit al despliegue sin drama", demostrando cómo:

- Partiendo de una infraestructura ya desplegada (Minikube + Localstack + Terraform),
- Se definieron los manifiestos YAML para la aplicación,
- Se configuró un pipeline automatizado en GitHub Actions con un runner local,
- Y se consiguió un despliegue funcional, reproducible y completamente automatizado de la aplicación sobre el clúster.

**En resumen:** De un git push a la rama main, la app se construye, despliega y queda accesible en http://midominio.local/app, conectándose a servicios simulados en Localstack.

### 9. Recursos:

```
~ kubectl get pods -A
NAMESPACE
                             NAME
                                                                                                            READY
                                                                                                                          STATUS
                                                                                                                                                RESTARTS
                                                                                                                                                                              AGE
                                                                                                           1/1
0/1
                             lahuella-app-deployment-58b9669cf4-llkn5
                                                                                                                                                                              48m
default
                                                                                                                          Running
                                                                                                                                                                              4h35m
ingress-nginx
                              ingress-nginx-admission-create-mwwq6
                                                                                                                          Completed
                                                                                                                                                0
                                                                                                            0/1
ingress-nginx
                             ingress-nginx-admission-patch-tqcqs
                                                                                                                          Completed
                                                                                                                                                                              4h35m
                                                                                                                                                                              4h35m
ingress-nginx
                             ingress-nginx-controller-9cc49f96f-2kc5k
                                                                                                                          Running
                                                                                                                                                 0
kube-system
                             coredns-66bc5c9577-2kdwz
                                                                                                            1/1
                                                                                                                          Running
                                                                                                                                                    (4h51m ago)
                                                                                                                                                                              4h56m
                                                                                                                                                 1
                                                                                                                                                    (4h51m ago)
(4h51m ago)
                             etcd-lahuella
                                                                                                                                                                              4h56m
kube-system
                                                                                                                          Running
kube-system
                             kube-apiserver-lahuella
                                                                                                            1/1
                                                                                                                          Running
                                                                                                                                                                              4h56m
kube-system
                             kube-controller-manager-lahuella
                                                                                                            1/1
                                                                                                                                                    (4h51m ago)
                                                                                                                                                                              4h56m
                                                                                                                          Runnina
kube-system
                             kube-proxy-fmbc8
                                                                                                                          Running
                                                                                                                                                    (4h51m ago)
                                                                                                                                                                              4h56m
                             kube-scheduler-lahuella
kube-system
                                                                                                            1/1
                                                                                                                          Runnina
                                                                                                                                                1 (4h51m ago)
                                                                                                                                                                              4h56m
                                                                                                                                                3 (4h50m ago)
                             storage-provisioner
                                                                                                                                                                              4h56m
kube-system
                                                                                                            1/1
                                                                                                                          Running
localstack
                             localstack-b8c864885-mnvz5
                                                                                                                          Running
                                                                                                                                                    (4h51m ago)
                                                                                                                                                                              4h54m
                             localstack-seed-gkfrq
                                                                                                                                                                              4h41m
localstack
                                                                                                                          Error
         kubectl get ingress -A
NAMESPACE
                       NAME
                                                                 CLASS
                                                                               HOSTS
                                                                                                                 ADDRESS
                                                                                                                                             PORTS
                                                                                                                                                           AGE
                        lahuella-app-ingress
                                                                                                                 192.168.49.2
default
                                                                 nginx
                                                                                midominio.local
                                                                                                                                             80
                                                                                                                                                           39m
                        localstack-ingress
localstack
                                                                 nginx
                                                                                midominio.local
                                                                                                                 192.168.49.2
                                                                                                                                             80
                                                                                                                                                           4h53m
      ~ curl http://midominio.local/_localstack/health
~ curl http://midominio.local/_localstack/health
{"services": {"acm": "available", "apigateway": "available", "cloudformation": "available", "cloudwatch":
"running", "config": "available", "dynamodb": "running", "dynamodbstreams": "available", "ec2": "available
", "es": "available", "events": "available", "firehose": "available", "iam": "available", "kinesis": "available", "kinesis": "available", "kinesis": "available", "lambda": "available", "logs": "running", "opensearch": "available", "redshift
": "available", "resource-groups": "available", "resourcegroupstaggingapi": "available", "route53": "available", "route53resolver": "available", "s3": "running", "s3control": "available", "scheduler": "available",
, "secretsmanager": "available", "ses": "available", "sns": "available", "sqs": "running", "ssm": "available",
purchase "available", "stepfunctions": "available", "sts": "available", "support": "available", "swf": "available", "transcribe": "available", "edition": "community", "version": "4.9.3.dev38"}
```

```
terraform git:(main) x terraform state list
aws cloudwatch log group.la huella aplication
aws dynamodb table.la huella analytics
aws dynamodb table.la huella comments
aws dynamodb table.la huella products
aws s3 bucket.la huella sentiment reports
aws s3 bucket.la huella uploads
aws sqs queue.la huella notifications queue
aws sqs queue.la huella processing queue
   terraform git: (main) x awslocal --endpoint-url=http://midominio.local dynamodb list-tables
{
    "TableNames": [
        "la-huella-analytics",
        "la-huella-comments"
        "la-huella-products"
  terraform git:(main) x
```

#### deployment-app.yaml

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: lahuella-app-deployment
  namespace: default
  labels:
    app: lahuella-app
spec:
  replicas: 1
  selector:
    matchLabels:
      app: lahuella-app
  template:
    metadata:
      labels:
        app: lahuella-app
```

```
spec:
  containers:
   - name: lahuella-app
      image: lahuella-app:latest
      imagePullPolicy: IfNotPresent
     ports:
        - containerPort: 3000
      env:
        - name: NODE ENV
         value: "production"
        - name: AWS ACCESS KEY ID
         value: "test"
        - name: AWS_SECRET ACCESS KEY
         value: "test"
        - name: AWS DEFAULT REGION
         value: "eu-west-1"
        - name: AWS ENDPOINT URL
         value: "http://midominio.local"
        - name: DYNAMODB TABLE PRODUCTS
          value: "la-huella-products"
        - name: DYNAMODB TABLE COMMENTS
         value: "la-huella-comments"
        - name: DYNAMODB TABLE ANALYTICS
          value: "la-huella-analytics"
        - name: S3 BUCKET REPORTS
         value: "la-huella-sentiment-reports"
        - name: S3 BUCKET UPLOADS
          value: "la-huella-uploads"
      readinessProbe:
        httpGet:
          path: /api/health
          port: 3000
        initialDelaySeconds: 10
        periodSeconds: 5
      livenessProbe:
       httpGet:
          path: /api/health
          port: 3000
        initialDelaySeconds: 20
        periodSeconds: 10
```

#### ingress-app.yaml:

```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
   name: lahuella-app-ingress
   namespace: default
   annotations:
    nginx.ingress.kubernetes.io/rewrite-target: /
spec:
```

```
ingressClassName: nginx
rules:
    - host: midominio.local
    http:
    paths:
        - path: /app
        pathType: Prefix
        backend:
        service:
        name: lahuella-app-service
        port:
        number: 80
```

### • service-app.yaml:

```
apiVersion: v1
kind: Service
metadata:
    name: lahuella-app-service
    namespace: default
    labels:
        app: lahuella-app
spec:
    selector:
        app: lahuella-app
ports:
    - protocol: TCP
        port: 80
        targetPort: 3000
type: ClusterIP
```

### • deploy.yaml:

```
name: CI/CD - Deploy to Minikube

on:
    push:
        branches:
        - main
        workflow_dispatch:

jobs:
    build-and-deploy:
    name: Build & Deploy App to Minikube
    runs-on: self-hosted
```

```
steps:
  - name: Checkout repository
   uses: actions/checkout@v4
  - name: Set up environment variables
   run:
     export AWS ACCESS KEY ID=test
     export AWS SECRET ACCESS KEY=test
     export AWS DEFAULT REGION=eu-west-1
     export AWS ENDPOINT URL=http://midominio.local
  - name: Build Docker image
   run:
     echo " Building Docker image for the app..."
     docker build -t lahuella-app:latest .
     echo " Image built successfully"
  - name: Load image into Minikube
   run:
     echo " Loading image into Minikube..."
     minikube -p lahuella image load lahuella-app:latest
  - name: Apply Kubernetes manifests
    run:
     echo " Applying Kubernetes manifests..."
     kubectl apply -f infra/k8s/deployment-app.yaml
     kubectl apply -f infra/k8s/service-app.yaml
     kubectl apply -f infra/k8s/ingress-app.yaml
     echo " Application deployed successfully!"
  - name: Verify running pods
   run:
     echo " Checking deployment status..."
     kubectl get pods -o wide
  - name: Test application URL
   run:
     kubectl get ingress
     echo "Try accessing http://midominio.local"
```