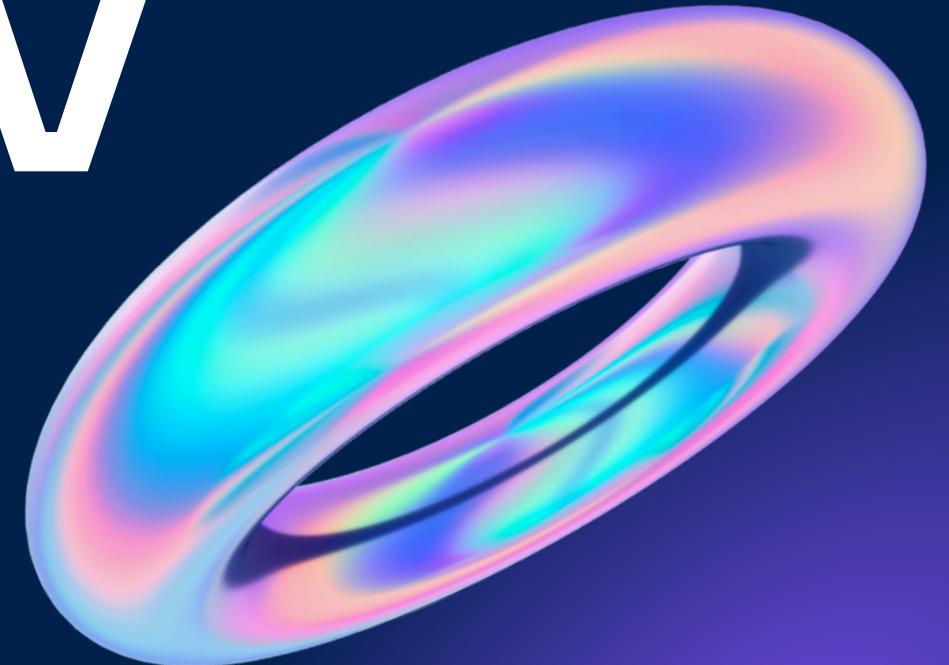




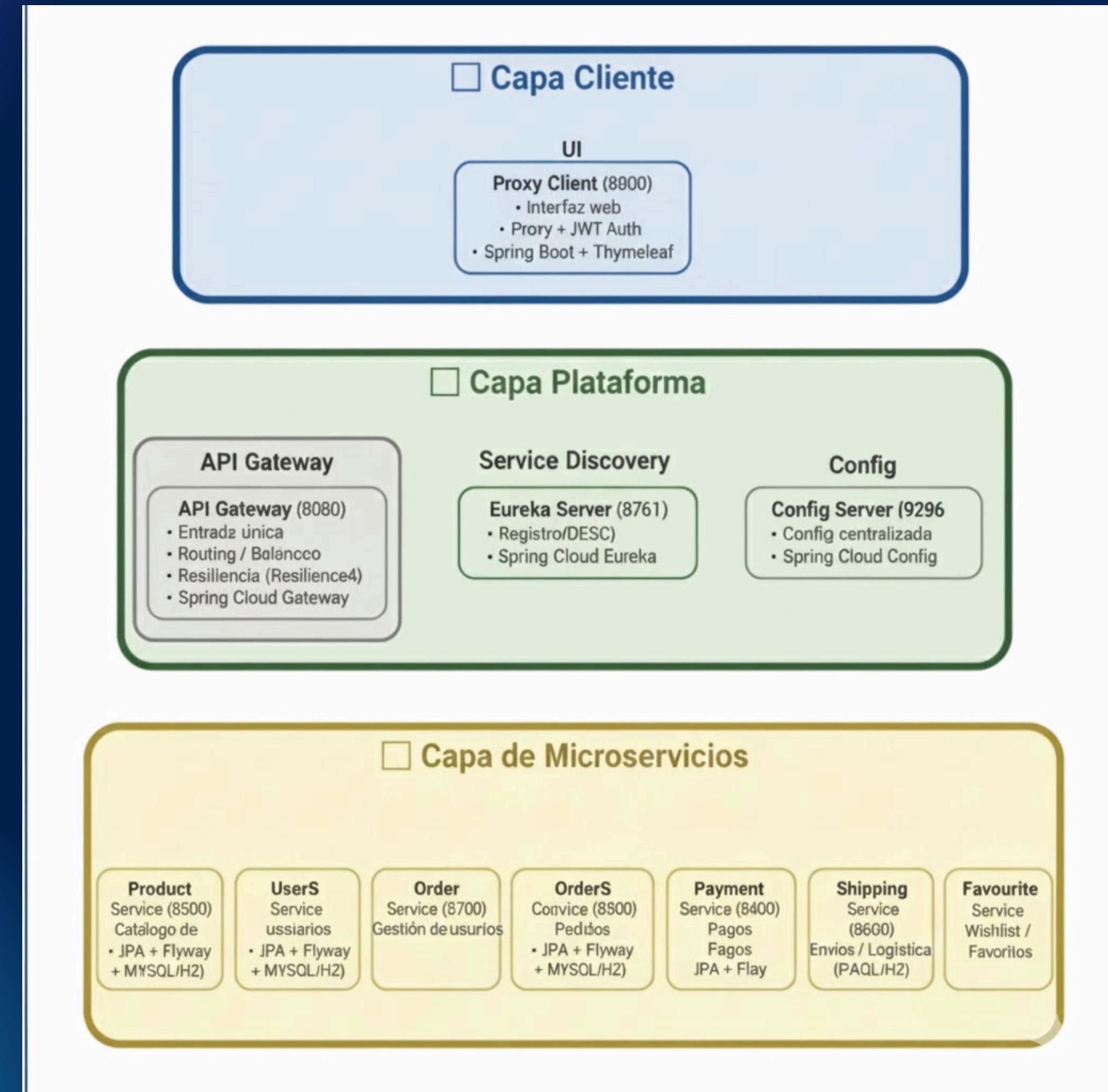
Entrega Final

Ingesoft V

Andrés Cabezas
Davide Flaminii



MICROSERVICIOS



METODOLOGÍA ÁGIL

Kanban

Permite visualizar el flujo de trabajo con un tablero dividido en columnas: Backlog, To Do, In Progress, In review, Done

davidone007 / Projects / Kanban - Entrega Final Ingesoft V

Type to search

Add status update

Insight

Kanban - Entrega Final Ingesoft V

Backlog | Priority board | Team items | Roadmap | + New view

Filter by keyword or by field

Backlog 0 / 11 Estimate: 0

This item hasn't been started

To Do 0 Estimate: 0

In progress 0 / 11 Estimate: 0

This is actively being worked on

In review 0 / 11 Estimate: 0

This item is in review

Done 11 Estimate: 0

This has been completed

HU3 – Iniciar ambiente (dev, stage, prod)

StockSync #4
HU4 – Backend remoto de Terraform

StockSync #5
HU5 – Despliegue de microservicios en Kubernetes

StockSync #6
HU6 – Configuración de probes y health checks

StockSync #7
HU7 – Pipeline de CI/CD (GitHub Actions)

StockSync #8
HU8 – Integración de análisis de seguridad y calidad

StockSync #9
HU9 – Implementación de pruebas completas

StockSync #10
HU10 – Monitoreo y Observabilidad

+ Add item

Historias de Usuario

- StockSync #1
HU1 – Configuración de Infraestructura con Terraform
- StockSync #2
HU2 – Modularización de la Infraestructura
- StockSync #3
HU3 – Multi-ambiente (dev, stage, prod)
- StockSync #4
HU4 – Backend remoto de Terraform
- StockSync #5

HU1 – Configuración de Infraestructura con Terraform #1

Closed davidone007/StockSync Public

 andrescabezas26 opened 5 days ago Collaborator ...

Como DevOps Engineer necesito aprovisionar toda la infraestructura del proyecto usando Terraform para garantizar que los ambientes sean reproducibles, escalables y fáciles de administrar.

Criterios de Aceptación

- Debe existir un directorio infra/ con los archivos principales (main.tf, variables.tf, outputs.tf).
- Ejecutar terraform init y terraform apply debe crear la infraestructura sin errores.
- La infraestructura creada debe estar documentada en el README del módulo.
- La ejecución debe ser reproducible en cualquier máquina con Terraform instalado.

Create sub-issue

 andrescabezas26 moved this to Backlog in Kanban - Entrega Final Ingesoft V 5 days ago

 andrescabezas26 added this to Kanban - Entrega Final Ingesoft V and Kanban - Entrega Final Ingesoft V and removed this from Kanban - Entrega Final Ingesoft V 5 days ago

 andrescabezas26 moved this to Backlog in Kanban - Entrega Final Ingesoft V 5 days ago

 andrescabezas26 moved this from Backlog to In progress in Kanban - Entrega Final Ingesoft V 5 days ago

Edit ... x

Assignees: No one - [Assign yourself](#) Assign to Copilot

Labels: No labels

Projects: Kanban - Entrega Final Ingesoft V

Status: Done

Priority: Choose an option

Size: Choose an option

Estimate: Enter number...

Start date: No date

Target date: No date

Milestone:

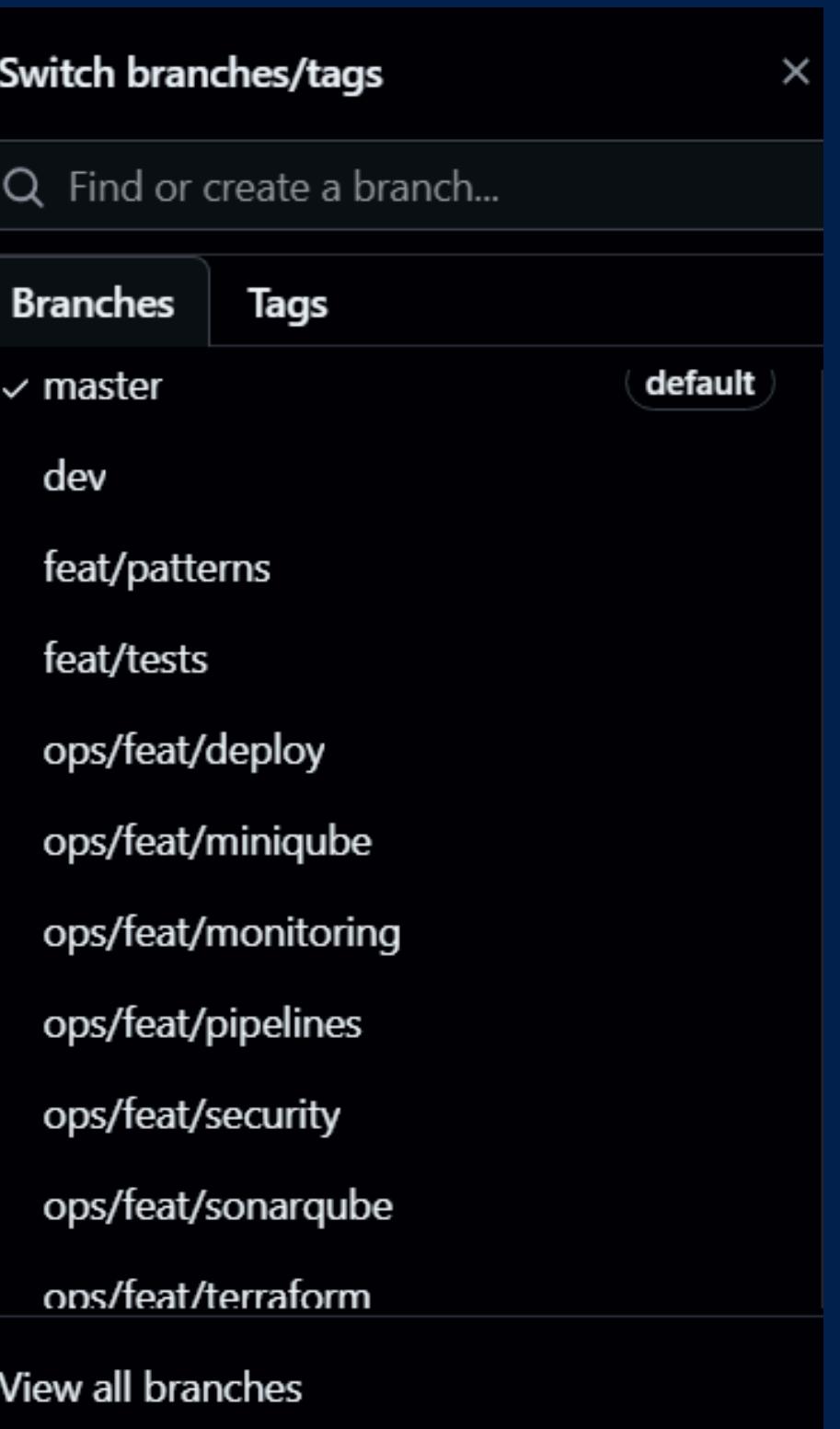
Estrategia de Branching

Git Flow Simplificado



Desarrolladores

- **main:** rama protegida, siempre lista para producción.
- **develop:** integración continua, tambien protegida.
- **feature/<ticket>:** nuevas funcionalidades o fixes menores.
- **hotfix/<issue>:** correcciones críticas.



Operaciones

- **ops/feature/<cambio>:** propuesta de cambio infra.
- **ops/hotfix/<issue>:** correcciones urgentes de infraestructura.

Protección de ramas

Branch protection rules

Add rule

master	Currently applies to 1 branch	Edit	Delete
stage	Currently applies to 1 branch	Edit	Delete
dev	Currently applies to 1 branch	Edit	Delete

- Requerir un pull request antes de hacer merge
- Requerir aprobaciones
- Requerir que los checks de estado pasen antes de hacer merge

Require a pull request before merging
When enabled, all commits must be made to a non-protected branch and submitted via a pull request before they can be merged into a branch that matches this rule.

Require approvals
When enabled, pull requests targeting a matching branch require a number of approvals and no changes requested before they can be merged.
Required number of approvals before merging: 1 ▾

Dismiss stale pull request approvals when new commits are pushed
New reviewable commits pushed to a matching branch will dismiss pull request review approvals.

Require review from Code Owners
Require an approved review in pull requests including files with a designated code owner.

Require approval of the most recent reviewable push
Whether the most recent reviewable push must be approved by someone other than the person who pushed it.

Require status checks to pass before merging
Choose which [status checks](#) must pass before branches can be merged into a branch that matches this rule. When enabled, commits must first be pushed to another branch, then merged or pushed directly to a branch that matches this rule after status checks have passed.

Require branches to be up to date before merging
This ensures pull requests targeting a matching branch have been tested with the latest code. This setting will not take effect unless at least one status check is enabled (see below).

INFRAESTRUCTURA

Infrastuctura Terraform

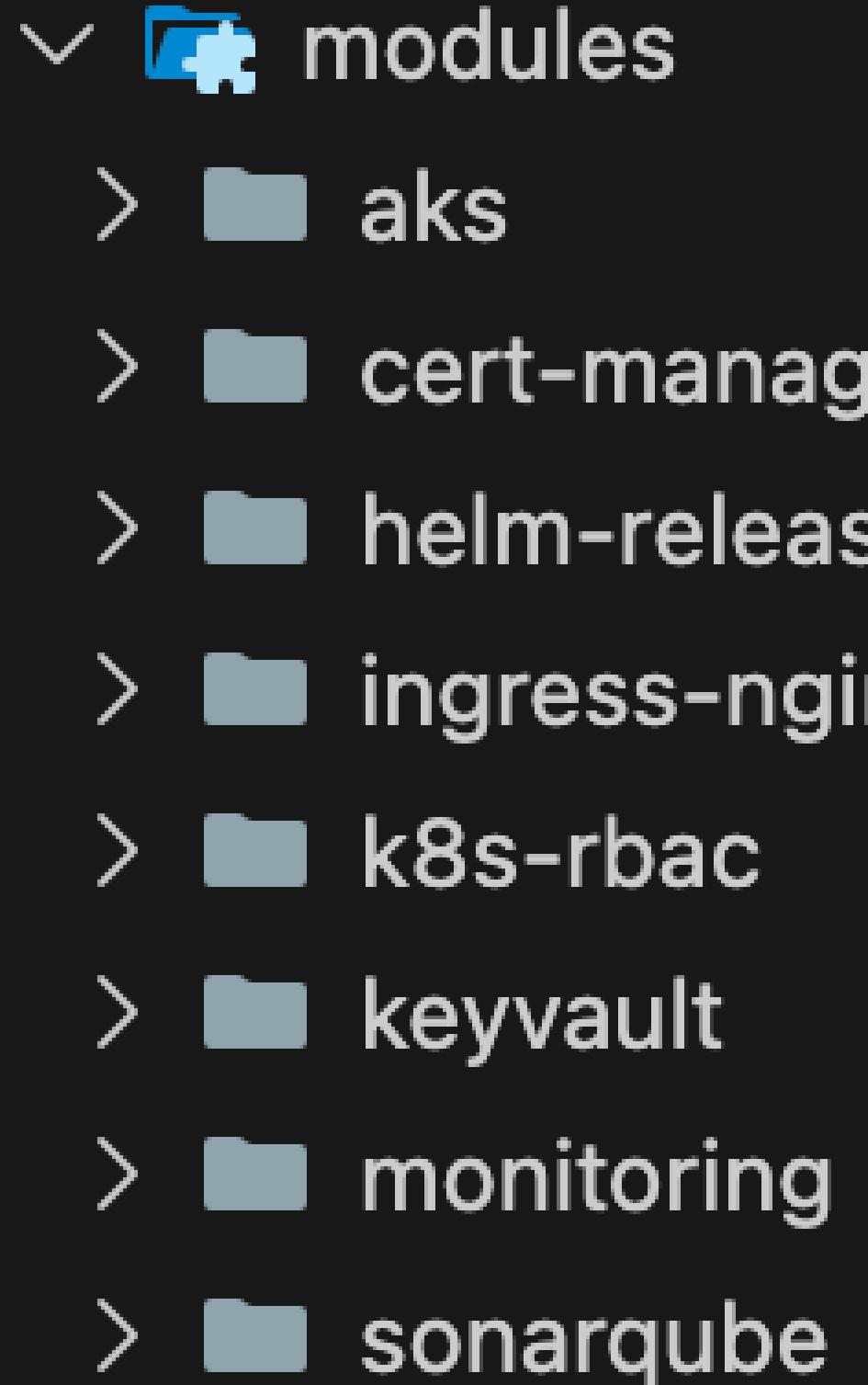
Backend remoto

<input type="checkbox"/>	Name ↑	Type	Location
<input type="checkbox"/>	tfstate40259	Storage account	East US

```
andrescabezas26, yesterday | 2 authors (You and one other)
terrafrom {
    andrescabezas26, yesterday | 2 authors (You and one other)
2     backend "azurerm" {
3         resource_group_name  = "tfstate-rg"
4         storage_account_name = "tfstate26906"
5         container_name       = "tfstate"
6         key                  = "stage.terraform.tfstate"
7     }
8 }
9
```

Infrastuctura Terraform

Modules



- **module aks:** Provisionar el clúster AKS y su Resource Group.
- **module keyvault:** Crear Key Vault y gestionar secretos.
- **module helm-release:** Desplegar aplicaciones y charts Helm.
- **module monitoring:** Instalar Prometheus, Grafana y ELK para observabilidad.
- **module ingress-nginx:** Instalar NGINX Ingress con IP estática.
- **module cert-manager:** Gestionar certificados con cert-manager y ClusterIssuer.
- **module k8s-rbac:** Crear roles y permisos RBAC en Kubernetes.
- **module sonarqube:** Desplegar SonarQube para análisis de calidad de código.

Infrastuctura Terraform

Nodes

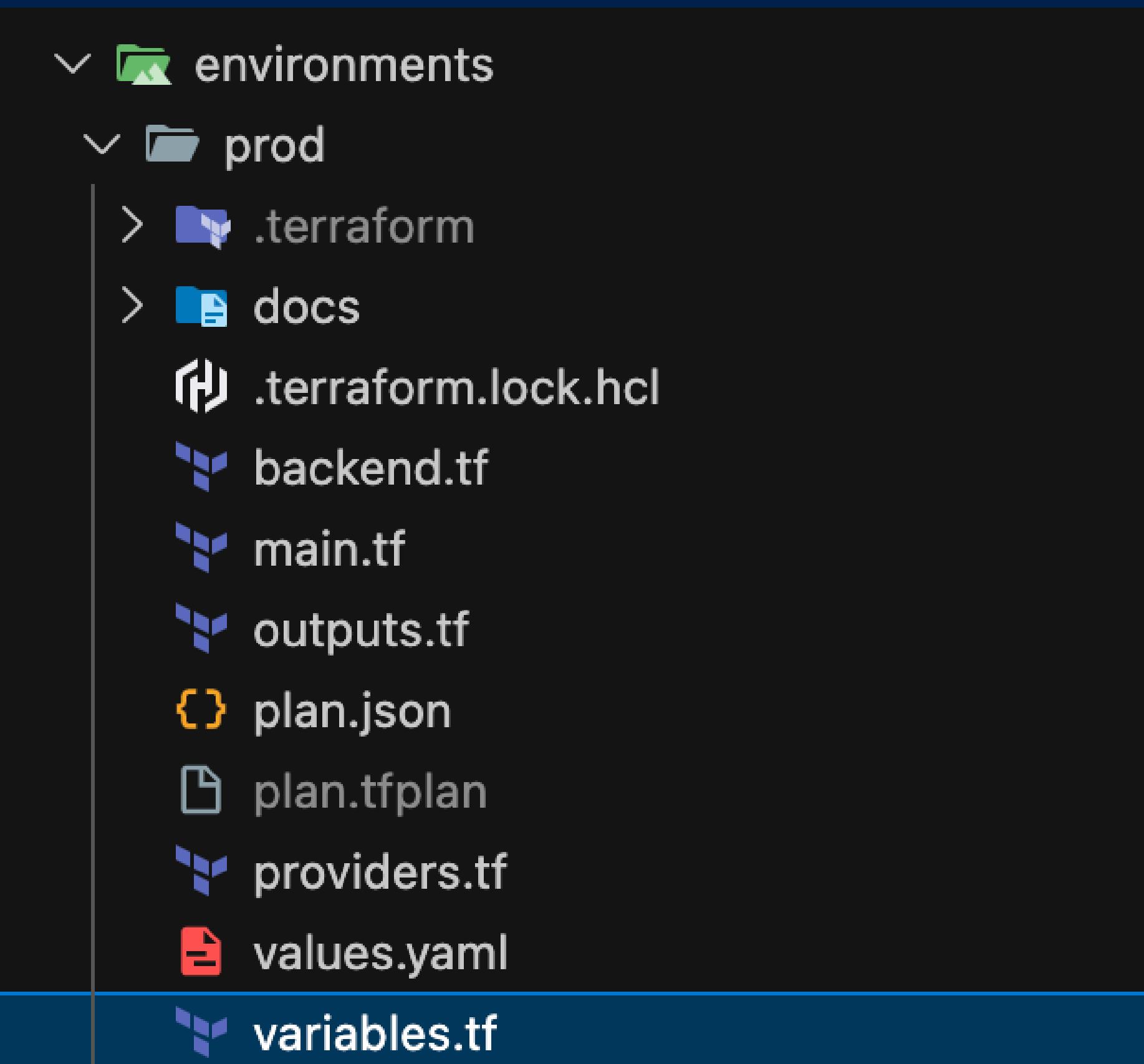
```
variable "node_count" {  
    description = "Number of nodes in the default node pool"  
    type        = number  
    default     = 2  
}
```

You, 5 days ago | 1 author (You)

```
variable "vm_size" {  
    description = "VM size for the nodes"  
    type        = string  
    default     = "Standard_E2_v3"  
}
```

Infrastuctura Terraform

Environments



Infrastuctura Terraform

ecommerce-static-resources   ... 

Help me generate Terraform for this resource group configuration. Generate Bicep code to duplicate this resource group. +1 X

Resource group

Search  Create  Manage view  Delete resource group  Refresh  Export to CSV  Open query  Assign tags  ... Group by none 

Overview  JSON View

Activity log  Access control (IAM)  Tags  Resource visualizer  Events  Settings 
Deployments  Security  Deployment stacks 

Essentials

Subscription (move)	: Azure for Students	Deployments	: No deployments
Subscription ID	: 256bea1e-51c0-4409-97c4-118ae42d16dc	Location	: East US
Tags (edit)	: Add tags		

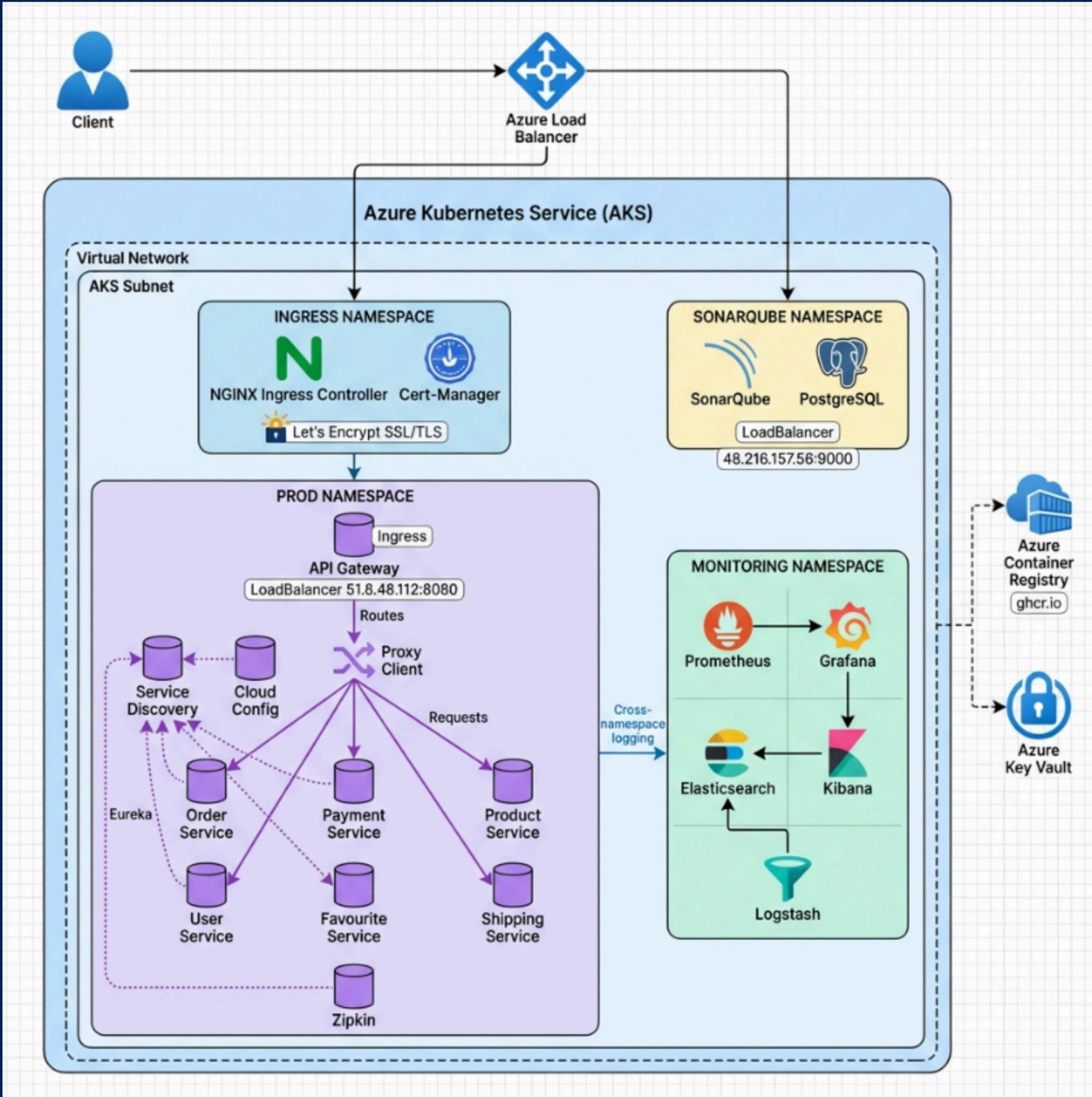
Resources  Recommendations

Filter for any field...  Type equals all  Location equals all  Add filter 

<input type="checkbox"/> Name ↑	Type	Location
<input type="checkbox"/>  ecommerce-prod-ip	... Public IP address	East US

IP ESTATICA PARA EL DEPLOY DE PRODUCCIÓN

Diagrama de Arquitectura



The background features a dark blue gradient with three glowing, translucent 3D shapes: a torus on the left, a large pyramid on the right, and a partial ring at the bottom left.

PATRONES DE DISEÑO

Patrones de Diseño

Hot Reload

¿Qué problema resuelve?

Permite:

- ✓ Cambiar config en tiempo real
- ✓ No quemar valores en el código
- ✓ Usar múltiples perfiles (dev/stage/prod)

Configuración del servicio

```
spring:  
  application:  
    name: product-service  
  cloud:  
    config:  
      uri: ${SPRING_CONFIG_IMPORT:localhost:9296}  
      fail-fast: false  
  profiles:  
    active: dev
```

Consumir propiedades externas desde el microservicio

```
@SpringBootApplication  
@EnableEurekaClient  
@EnableConfigServer // ← Activa el servidor de configuración  
public class CloudConfigApplication {  
  public static void main(String[] args) {  
    SpringApplication.run(CloudConfigApplication.class, args);  
  }  
}
```

Patrones de Diseño

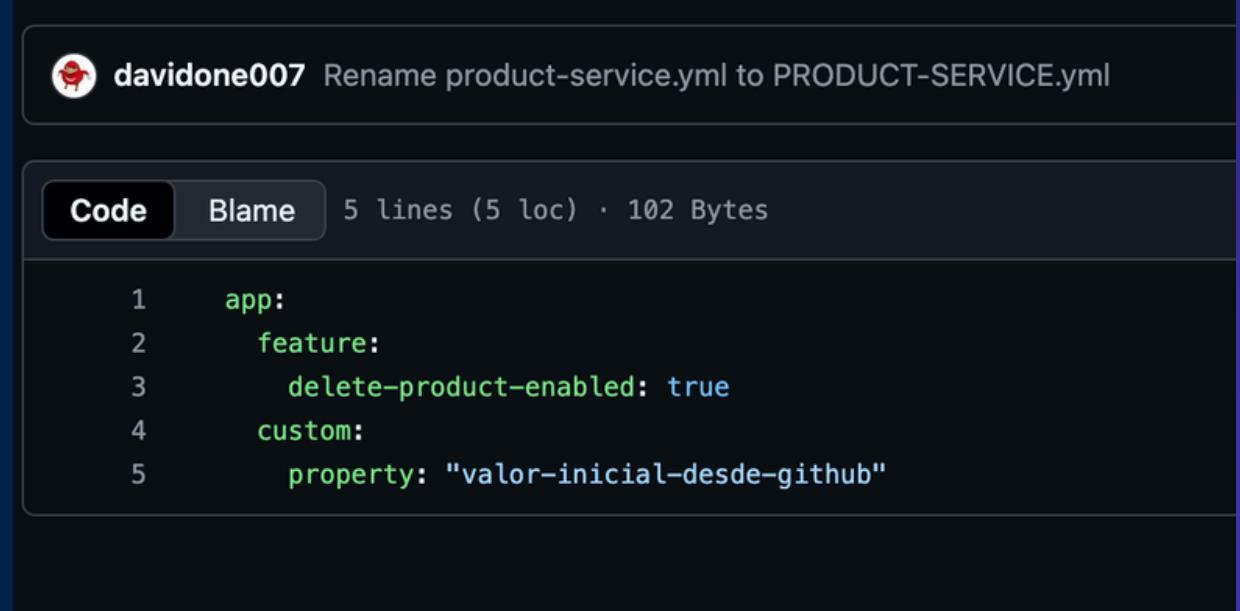
Hot Reload

Consumir propiedades externas desde el microservicio

```
@RestController
@RequestMapping("/api/products")
@RefreshScope // ← Permite refresh sin reiniciar
public class ProductResource {

    @Value("${app.custom.property:default}")
    private String customProperty;

    @GetMapping("/config/custom-property")
    public ResponseEntity<String> getCustomProperty() {
        return ResponseEntity.ok(this.customProperty);
    }
}
```



```
1 app:
2   feature:
3     delete-product-enabled: true
4   custom:
5     property: "valor-inicial-desde-github"
```

Demostración del Hot Reload

• Cambias la propiedad en GitHub

• Llamas al refresh:

```
VALOR-SUPER-NUEVO-Z0Z4
> curl -X POST http://localhost:8500/product-service/actuator/refresh
[ "config.client.version", "app.custom.property" ]%
< curl -X POST http://localhost:8500/product-service/actuator/refresh
```

- El endpoint /config/custom-property muestra el valor nuevo SIN reiniciar.

Patrones de Diseño

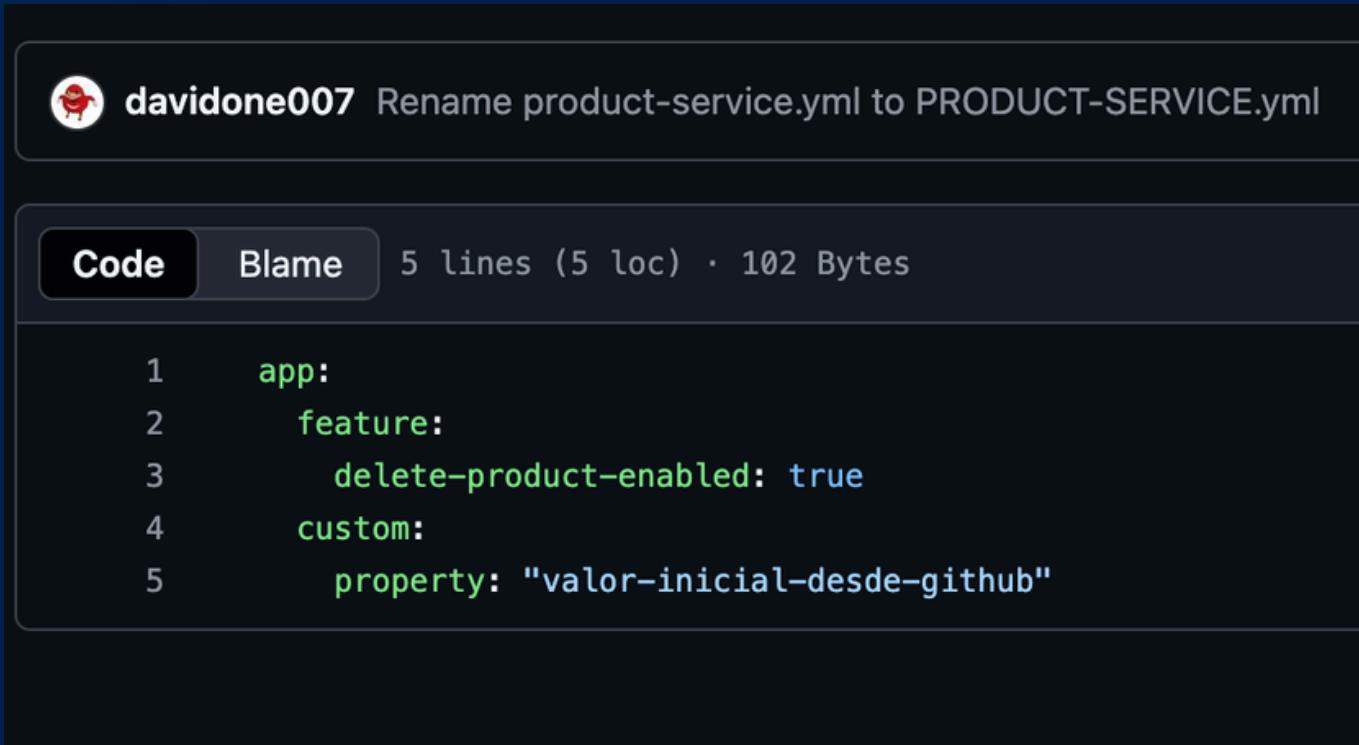
Feature Toggle

Un Feature Toggle permite activar o desactivar funcionalidades en tiempo real sin necesidad de redeploy.

Sirve para:

- ✓ Pruebas A/B
- ✓ Activar features gradualmente
- ✓ Controlar riesgos
- ✓ Hotfix sin despliegue

Configuración externa en GitHub (config-server)



davidone007 Rename product-service.yml to PRODUCT-SERVICE.yml

Code Blame 5 lines (5 loc) · 102 Bytes

```
1 app:
2   feature:
3     delete-product-enabled: true
4   custom:
5     property: "valor-inicial-desde-github"
```

Inyección dinámica en Spring Boot

```
@RestController
@RequestMapping("/api/products")
@RefreshScope
public class ProductResource {

    @Value("${app.feature.delete-product-enabled:true}")
    private boolean deleteProductEnabled;

    @DeleteMapping("/{productId}")
    public ResponseEntity<Boolean> deleteById(
        @PathVariable("productId") final String productId) {

        if (!this.deleteProductEnabled) {
            log.info("**** Feature Toggle: Delete product is disabled ***");
            return ResponseEntity.status(403).body(false);
        }

        log.info("**** Delete product by id ***");
        this.productService.deleteById(Integer.parseInt(productId));
        return ResponseEntity.ok(true);
    }
}
```

Actualización en caliente después de editar en GitHub

```
> curl http://localhost:8500/product-service/api/products/config/custom-property
cambiando-falso%
```

Patrones de Diseño

Feature Toggle

Un Feature Toggle permite activar o desactivar funcionalidades en tiempo real sin necesidad de redeploy.

Sirve para:

- ✓ Pruebas A/B
- ✓ Activar features gradualmente
- ✓ Controlar riesgos
- ✓ Hotfix sin despliegue

Cambiando la propiedad a falso en el endpoint /config/custom-property

```
> curl http://localhost:8500/product-service/api/products/config/custom-property  
cambiando-falso
```

Propiedad en falso

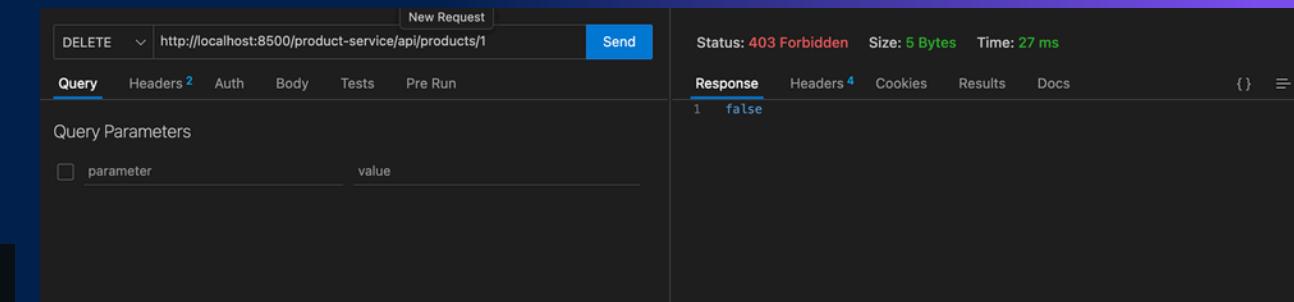
cloud-config-server / product-service.yml

davidone007 Update product-service.yml

Code Blame 5 lines (5 loc) · 92 Bytes

```
1 app:  
2   feature:  
3     delete-product-enabled: false  
4   custom:  
5     property: "cambiando-falso"
```

Intentar ejecutar el delete con la propiedad seteada como falso



Patrones de Diseño

Circuit Breaker

Protege el microservicio evitando que siga llamando a un servicio que está fallando y causando más daño.

Permite detener las llamadas automáticamente y reaccionar rápido ante fallas externas.

- ✓ Evitar cascadas de fallos entre microservicios
- ✓ Reducir tiempos de espera cuando un servicio está caído o lento
 - ✓ Aumentar la resiliencia del sistema completo
- ✓ Responder con fallbacks controlados mientras la falla se resuelve
- ✓ Restablecer las llamadas automáticamente cuando el servicio se recupera

Configuración del Circuit Breaker – Resilience4j

```
resilience4j:  
  circuitbreaker:  
    instances:  
      productService:  
        register-health-indicator: true  
        event-consumer-buffer-size: 10  
        automatic-transition-from-open-to-half-open-enabled: true  
        failure-rate-threshold: 50          # Abre si 50% de requests fallan  
        minimum-number-of-calls: 5          # Mínimo 5 llamadas  
        permitted-number-of-calls-in-half-open-state: 3 # 3 intentos en HALF_OPEN  
        sliding-window-size: 10            # Ventana de 10 llamadas  
        wait-duration-in-open-state: 5s     # Espera 5s antes de HALF_OPEN  
        sliding-window-type: COUNT_BASED
```

Estadísticas de Health Check

```
management:  
  endpoints:  
    web:  
      exposure:  
        include: "*" # Expone todos los endpoints  
  health:  
    circuitbreakers:  
      enabled: true # Monitorea circuit breakers  
    endpoint:  
      health:  
        show-details: always # Muestra detalles en /actuator/health
```

Patrones de Diseño

Circuit Breaker

Protege el microservicio evitando que siga llamando a un servicio que está fallando y causando más daño.

Permite detener las llamadas automáticamente y reaccionar rápido ante fallas externas.

- ✓ Evitar cascadas de fallos entre microservicios
- ✓ Reducir tiempos de espera cuando un servicio está caído o lento
 - ✓ Aumentar la resiliencia del sistema completo
- ✓ Responder con fallbacks controlados mientras la falla se resuelve
 - ✓ Restablecer las llamadas automáticamente cuando el servicio se recupera

Replicas del servicio seteadas a 0, imposibilitando la peticiones al servicio

Llamada al endpoint en estado Cerrado

```
> curl http://localhost:8080/product-service/api/products
{
  "collection": [
    {
      "productId": 1,
      "productTitle": "asus",
      "imageUrl": "xxx",
      "sku": "dfqejklejrkn",
      "priceUnit": "0.0",
      "quantity": 50,
      "category": {
        "categoryId": 1,
        "categoryTitle": "Computer",
        "imageUrl": null
      }
    },
    {
      "productId": 2,
      "productTitle": "hp",
      "imageUrl": "xxx",
      "sku": "zsejfedbjh",
      "priceUnit": "0.0",
      "quantity": 50,
      "category": {
        "categoryId": 1,
        "categoryTitle": "Computer",
        "imageUrl": null
      }
    },
    {
      "productId": 3,
      "productTitle": "Armani",
      "imageUrl": "xxx",
      "sku": "fjdvf",
      "priceUnit": "0.0",
      "quantity": 50,
      "category": {
        "categoryId": 2,
        "categoryTitle": "Mode",
        "imageUrl": null
      }
    }
  ]
}
```

Llamada al endpoint con estado Abierto

```
> curl -v http://localhost:8080/product-service/api/products
* Trying 127.0.0.1:8080...
* Connected to localhost (127.0.0.1) port 8080 (#0)
> GET /product-service/api/products HTTP/1.1
> Host: localhost:8080
> User-Agent: curl/8.1.2
> Accept: */*
>
< HTTP/1.1 503 Service Unavailable
< Vary: Origin
< Vary: Access-Control-Request-Method
< Vary: Access-Control-Request-Headers
< Content-Type: text/plain; charset=UTF-8
< Content-Length: 65
<
* Connection #0 to host localhost left intact
Product Service is currently unavailable. Please try again later.%
```

```
> kubectl scale deployment product-service-container --replicas=0 -n stage
deployment.apps/product-service-container scaled
```

The background features a dark blue gradient with three glowing, translucent 3D rings in the foreground. One ring is positioned above the text, another is at the bottom left, and a third is partially visible on the right side. A large, semi-transparent prism is located on the right, with its facets reflecting a spectrum of colors.

CI/CD AVANZADO

Pipelines CI/CD

CI (Integración Continua)

Compila el código, corre pruebas unitarias, hace análisis básico y despliega automáticamente al ambiente de desarrollo para validar cambios rápidos del equipo.

Actions

All workflows

CI - CD - Dev build

CI - CD - Master build (Release + K8s de...

CI - CD - Stage build (Integration tests ...

New workflow

← CI - CD - Dev build

feat: Add pull request support for CI/CD tests on stage/master and re... #70

Re-run all jobs ⋮

Summary

Triggered via push 2 hours ago
davidone007 pushed → 758e56d ops/feat/deploy

Status: Success | Total duration: 6m 44s | Artifacts: 3

ci-cd-dev.yml
on: push

Jobs

- Build (Maven)
- Calculate Semantic Version (Dev)
- Repository filesystem & config s...
- Build api-gateway
- Build cloud-config
- Build favourite-service
- Build order-service
- Build payment-service
- Build product-service
- Build proxy-client
- Build service-discovery
- Build shipping-service
- Build user-service
- Generate Dev Release Notes
- Notify Success
- Notify Failure or Cancellation

Artifacts

Produced during runtime

Name	Size	Digest
test-results	33.9 KB	sha256:07705030f1f27a6a281b037be6d37cbffd02d139139...

[Minimize] [Maximize] [Close]

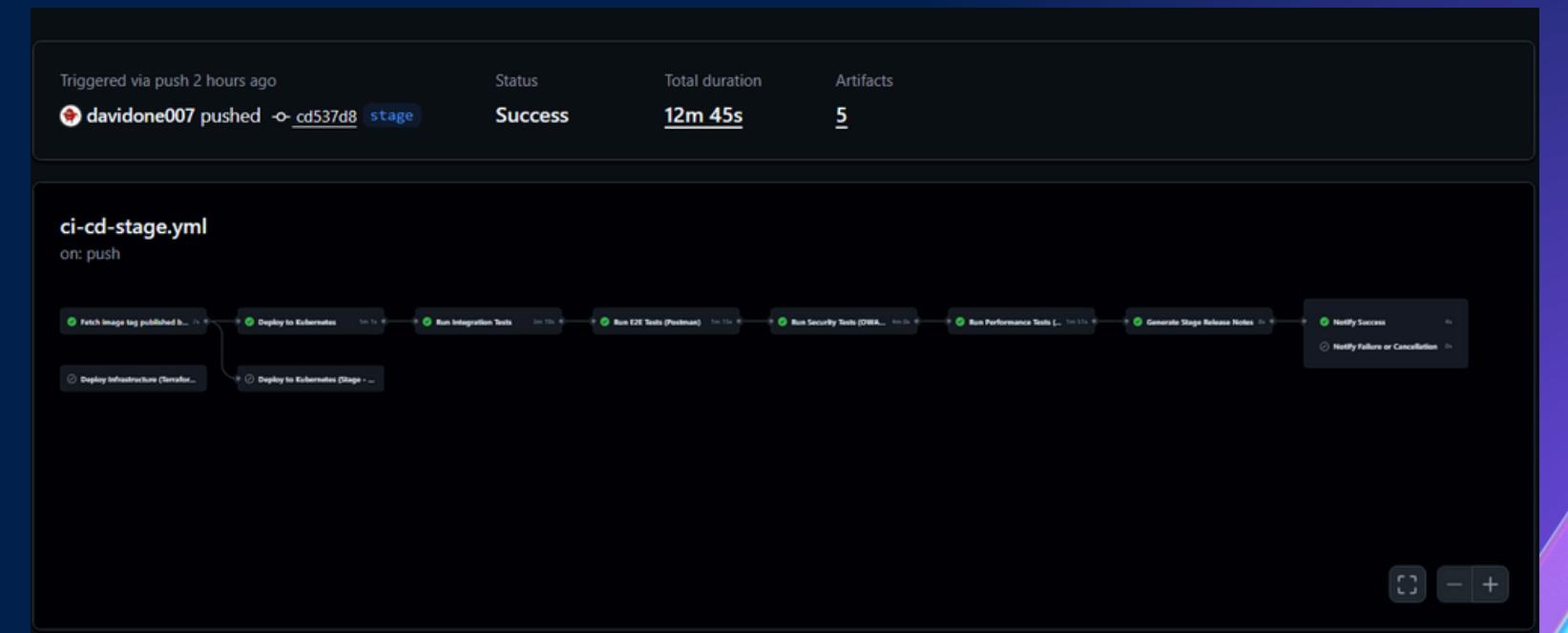
Pipelines CI/CD

CD (Despliegue Continuo)

Stage

Ejecuta pruebas de integración, E2E, seguridad y rendimiento. Si todo pasa, despliega en stage para validar antes de producción.

Jobs
Fetch image tag published by Dev
Deploy Infrastructure (Terraform)
Deploy to Kubernetes
Deploy to Kubernetes (Stage - Inf...)
Run Integration Tests
Run E2E Tests (Postman)
Run Security Tests (OWASP ZAP)
Run Performance Tests (Locust)
Generate Stage Release Notes
Notify Success
Notify Failure or Cancellation



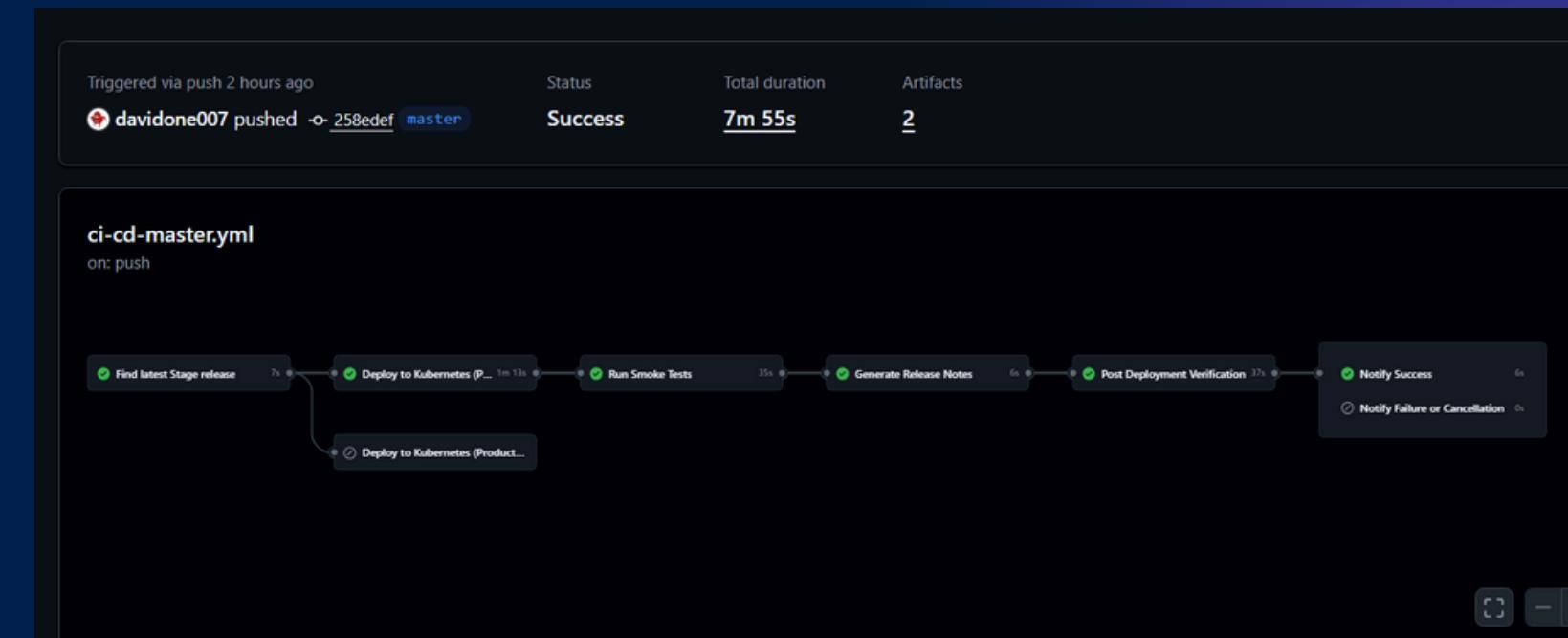
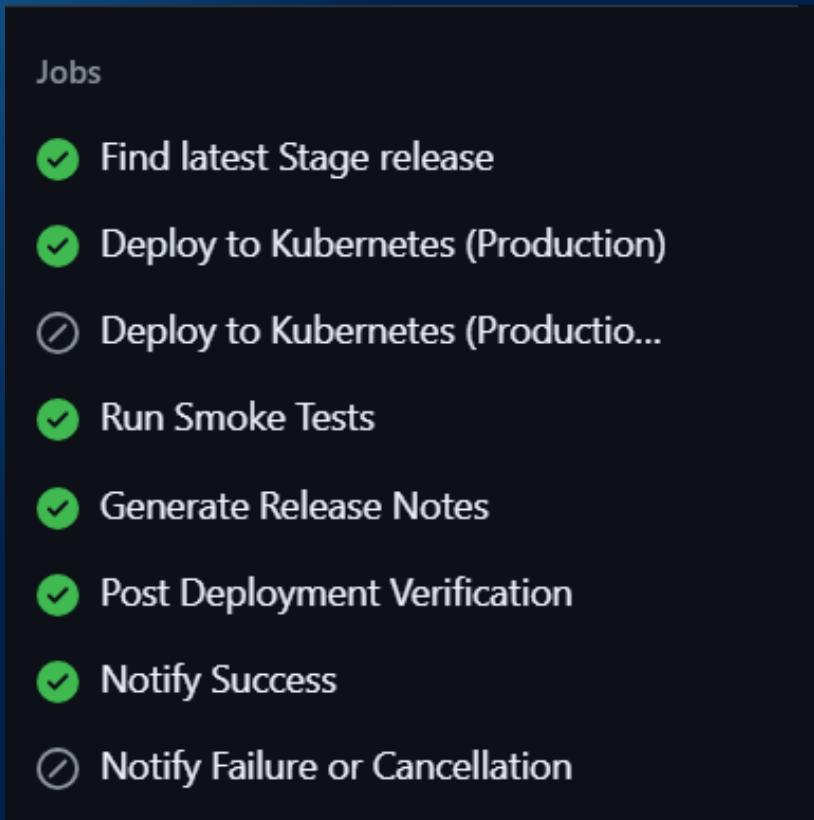
Artifacts			
Produced during runtime			
e2e-test-report	110 KB	sha256:8df8f19d3fc26515056ebdc49173a4eb284f7c2e9...	Download Delete
integration-test-results	34.5 KB	sha256:3f8684e1a34f2afbcfef5b1ce40d1f8e2b8b1b19739...	Download Delete
performance-test-results	1.97 MB	sha256:c84075d659e6f6088c646205e52960594ed15f4e7de...	Download Delete
stage-release-notes	1.13 KB	sha256:f5e87bbfd8b7ab187f30d37d9c01c1ac03631c5025...	Download Delete
zap-security-report	23.1 KB	sha256:172ab150fef78ac664fa6a2ffcc0335a4826921eb1...	Download Delete

Pipelines CI/CD

CD (Despliegue Continuo)

Prod / Master

Toma el artefacto aprobado, realiza validaciones de seguridad y despliega de forma controlada en producción, asegurando estabilidad y mínima afectación a usuarios.



Artifacts		
Produced during runtime		
Name	Size	Digest
release-notes	1.31 KB	sha256:91d5ca5076a5c452b033edb10d625775436d8b81011... Copy
smoke-test-reports	229 Bytes	sha256:a6d4863a5f085c67fcfa1b50ea478e547f4782822e15... Copy

Pipelines CI/CD

Notificaciones

The screenshot shows an email from GitHub Actions regarding a pipeline success. The subject is "Pipeline Success: davidone007/ecommerce-microservice-backend-app (refs/heads/stage)". The message body contains a link to the repository and commit details, and a note about the message being in English with a "Traducir al español" button. The email is from "GitHub Actions <ecommerce.microservice.ingesoftv@gmail.com>" and was sent at 7:13 p.m. (3 hours ago). It includes standard reply and forward buttons.

Ambientes

This screenshot shows the "Environments" section of the GitHub settings. It lists five environments: "stage-infra" (1 protection rule), "production_infra" (1 protection rule), "dev" (no protection rules), "stage" (2 protection rules), and "production" (1 protection rule). A "New environment" button is visible at the top right.

Aprobaciones

This screenshot shows the "Environments / Configure production" page. It displays deployment protection rules for the "production" environment. Under "Required reviewers", two users are listed: "davidone007" and "andrescabezas26". Other options like "Prevent self-review" and "Wait timer" are also shown. A "Save protection rules" button is at the bottom.

This screenshot shows a notification card for a workflow run. It says "[davidone007/ecommerce-microservice-backend-app] CI - CD - Master build (Release + K8s deploy + Smoke tests) workflow run". It indicates that the "CI - CD - Master build (Release + K8s deploy + Smoke tests): production is waiting for your review". A "Review pending deployments" button is present. Another section below shows "CI - CD - Master build (Release + K8s deploy + Smoke tests) / Deploy to Kubernetes (Production) production waiting for review".



PRUEBAS COMPLETAS

Pruebas unitarias

Se definieron pruebas unitarias para los microservicios del negocio

4036	[INFO] ecommerce-microservice-backend	SUCCESS [44.469 s]
4037	[INFO] service-discovery	SUCCESS [23.465 s]
4038	[INFO] cloud-config	SUCCESS [23.522 s]
4039	[INFO] api-gateway	SUCCESS [23.553 s]
4040	[INFO] proxy-client	SUCCESS [23.902 s]
4041	[INFO] user-service	SUCCESS [16.580 s]
4042	[INFO] product-service	SUCCESS [16.441 s]
4043	[INFO] favourite-service	SUCCESS [16.995 s]
4044	[INFO] order-service	SUCCESS [16.056 s]
4045	[INFO] shipping-service	SUCCESS [7.060 s]
4046	[INFO] payment-service	SUCCESS [6.724 s]
4047	[INFO] -----	
4048	[INFO] BUILD SUCCESS	

Validan la lógica de cada componente aislado.

Herramientas usadas: JUnit 5, Mockito.

Pruebas de integración

Se definieron pruebas integración para la comunicación entre microservicios

```
9394 [INFO] -----
9395 [INFO] Reactor Summary for ecommerce-microservice-backend 0.1.0:
9396 [INFO]
9397 [INFO] ecommerce-microservice-backend ..... SUCCESS [ 32.829 s]
9398 [INFO] service-discovery ..... SUCCESS [ 15.907 s]
9399 [INFO] cloud-config ..... SUCCESS [ 15.971 s]
9400 [INFO] api-gateway ..... SUCCESS [ 16.017 s]
9401 [INFO] proxy-client ..... SUCCESS [ 16.399 s]
9402 [INFO] user-service ..... SUCCESS [ 5.756 s]
9403 [INFO] product-service ..... SUCCESS [01:21 min]
9404 [INFO] favourite-service ..... SUCCESS [01:19 min]
9405 [INFO] order-service ..... SUCCESS [01:20 min]
9406 [INFO] shipping-service ..... SUCCESS [01:10 min]
9407 [INFO] payment-service ..... SUCCESS [ 24.447 s]
9408 [INFO] -----
9409 [INFO] BUILD SUCCESS
9410 [INFO] -----
9411 [INFO] Total time: 02:33 min (Wall Clock)
9412 [INFO] Finished at: 2025-11-26T00:05:39Z
9413 [INFO] -----
```

Características:

Validan interacción entre componentes y persistencia (DB, repositorios).

Herramientas: JUnit 5, Spring Boot Test, Testcontainers

@AutoConfigureTestDatabase(replace = AutoConfigureTestDatabase.Replace.NONE) (usa DB real o Testcontainers)

Uso de -Dgateway.url en CI para apuntar a la API gateway del entorno Stage

Pruebas E2E

Se definieron pruebas E2E para la validación de todo el flujo

- ✓ E2E Ecommerce Microservices Tests
 - > E2E-00: Setup Autenticación
 - > E2E-01: Registro y Gestión de Usuario Completa
 - > E2E-02: Flujo de Compra Completo
 - > E2E-03: Gestión de Productos y Stock
 - > E2E-04: Flujo de Envío Completo
 - > E2E-05: Gestión de Favoritos de Usuario
 - > E2E-99: Cleanup - Eliminar Datos de Prueba

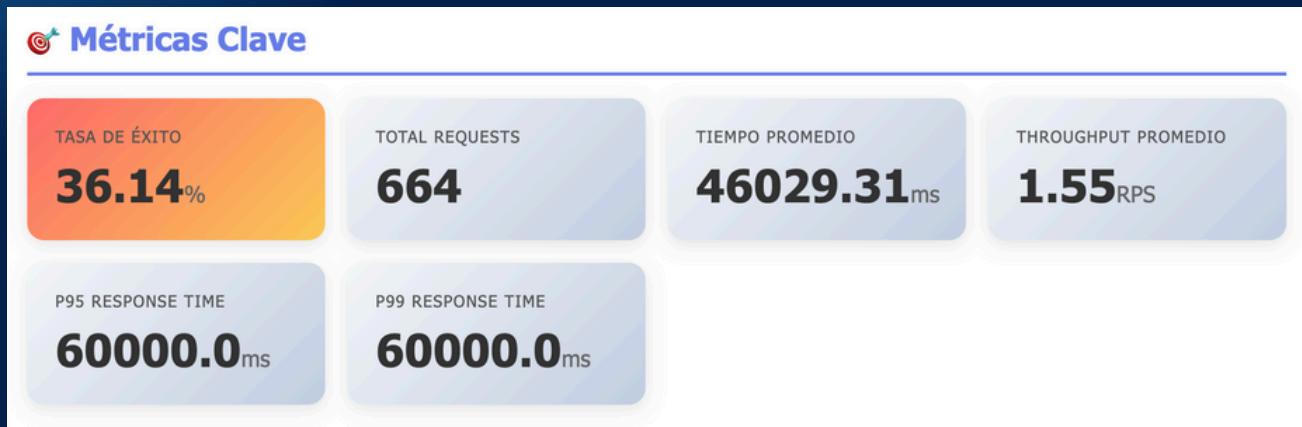
E2E Ecommerce Microservices... - Run results						
Source	Environment	Iterations	Duration	All tests	Avg. Resp. Time	
 Runner	none	1	28s 391ms	70	658 ms	
All Tests Passed (70) Failed (0) Skipped (0)						

Flujos completos de usuario a través del API Gateway (compras, pagos, consultas).

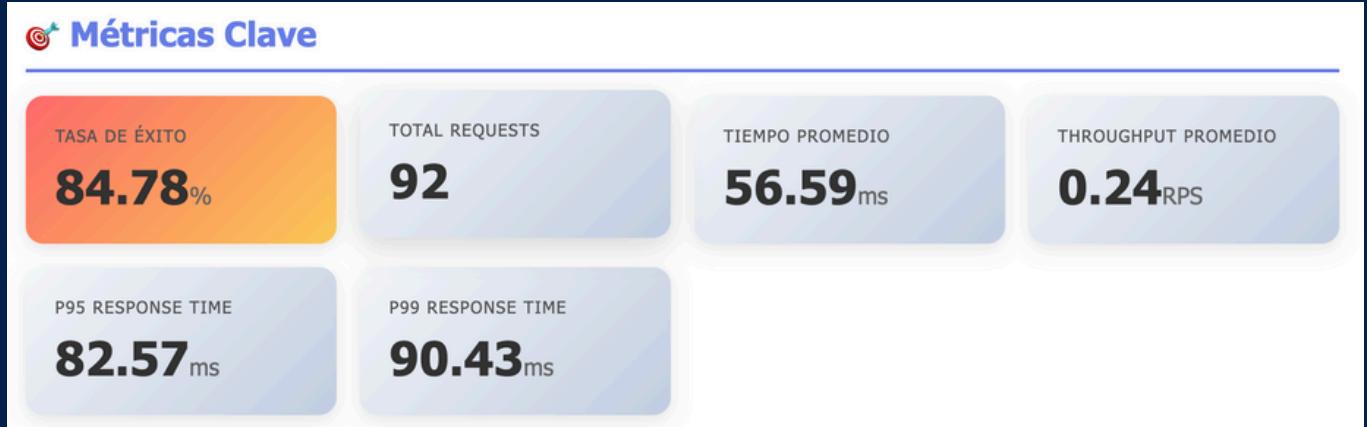
Herramientas: Postman Collections, Newman

Pruebas de rendimiento

Spike test



Quick start



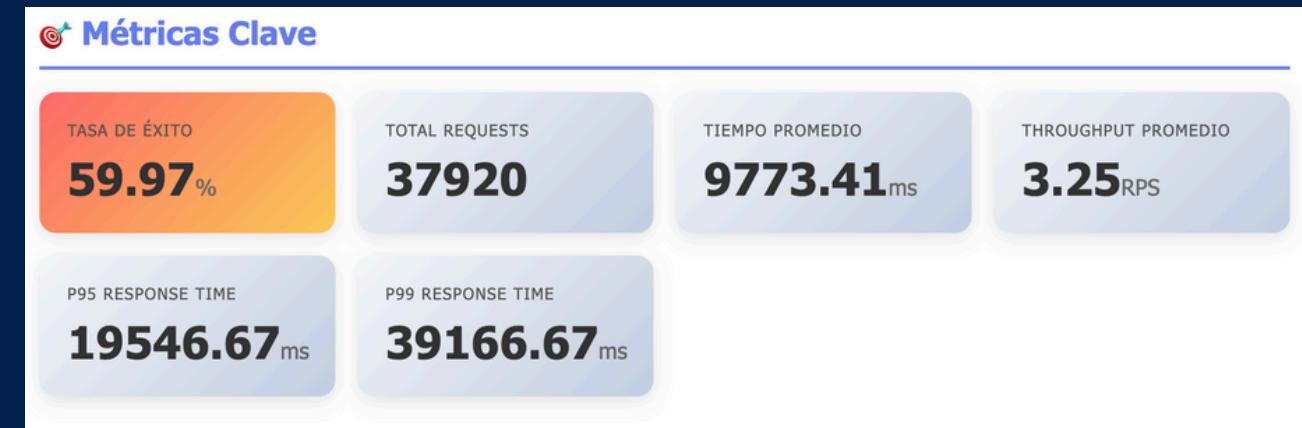
Load test



Soak test



Stress test



- **Quick Start:** 30s, 3 usuarios — prueba rápida
- **Load Test:** 5 min (ejemplo), 50 usuarios, spawn 5 — carga normal
- **Spike Test:** 3 min, 300 usuarios — pico rápido (alta rampa para bursts).
- **Stress Test:** 10–15 min, 200 usuarios — llevar al límite;
- **Soak Test:** 30 min, 100 usuarios — resistencia prolongada (detecta fugas y degradación).

Pruebas de seguridad

```
✓ OWASP ZAP Full Scan

1 ► Run zaproxy/action-full-scan@v0.10.0
16 Warning: : 'artifact_name' action input should not be empty. Setting it back to the default name.
17 starting the program
18 github run id :19687826889
19 /usr/bin/touch report_json.json report_md.md report_html.html
20 /usr/bin/chmod a+w report_json.json report_md.md report_html.html
21 /usr/bin/docker pull ghcr.io/zaproxy/zaproxy:stable -q
22 ghcr.io/zaproxy/zaproxy:stable
23 /usr/bin/docker run -v /home/runner/work/ecommerce-microservice-backend-app/ecommerce-microservice:/app ZAP_AUTH_HEADER_VALUE -e ZAP_AUTH_HEADER_SITE -t ghcr.io/zaproxy/zaproxy:stable zap-full-scan.py -r report_html.html -a
24 Total of 4 URLs
25 PASS: Directory Browsing [0]
26 PASS: Vulnerable JS Library (Powered by Retire.js) [10003]
27 PASS: Suspicious Input Transformation [100044]
28 PASS: In Page Banner Information Leak [10009]
29 PASS: Cookie No HttpOnly Flag [10010]
```

```
✓ ✓ Run OWASP Dependency-Check (Maven)

1 ► Run # Run dependency-check across the multi-module Maven project
9 [INFO] Scanning for projects...
10 [INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-dependencies/2.5.7.pom
11 [INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-dependencies/2.5.7.pom (8.6 kB at 39 kB/s)
12 [INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-dependencies/2020.0.4.pom
13 [INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/boot/spring-boot-dependencies/2020.0.4.pom (109 kB at 1.4 MB/s)
14 [INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/cloud/spring-cloud-dependencies-parent-3.0.4.pom
15 [INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/cloud/spring-cloud-dependencies-parent-3.0.4.pom (11 kB at 153 kB/s)
16 [INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/cloud/spring-cloud-dependencies-parent-3.0.4.pom
```

DAST (Dynamic Application Security Testing):

Herramienta: OWASP ZAP

Config: CI job security-tests (Stage) ejecuta zaproxy/action-full-scan@v0.10.0
apuntando a http://\$EXTERNAL_IP:8080

Resultado: genera report_html.html, report_json.json (artefactos subidos)

Dependency Scan (SCA):

Herramienta: OWASP Dependency-Check maven plugin (org.owasp:dependency-check-maven) en security-scan-schedule.yml.

Pruebas de humo

Se definieron pruebas de humo para la verificación del funcionamiento del sistema

```
1  ► Run echo "🔥 Running Smoke Tests..."  
67  🔥 Running Smoke Tests...  
68  
69  1 Testing API Gateway - Products...  
70  ✓ API Gateway - Products  
71  2 Testing Product Service...  
72  ✓ Product Service  
73  3 Testing User Service...  
74  ✓ User Service  
75  4 Testing Order Service...  
76  ✓ Order Service  
77  5 Testing Payment Service...  
78  ✓ Payment Service  
79  6 Testing Shipping Service...  
80  ✓ Shipping Service  
81  7 Testing Favourite Service...  
82  ✓ Favourite Service  
83  
84  📈 Smoke Test Results: 7 passed, 0 failed  
85  ✓ All smoke tests passed!
```

OBSERVABILIDAD Y MONITOREO



Monitoreo

Configuración de Actuator

Incluye Micrometer con el registro de Prometheus para exportar métricas en formato que Prometheus puede consumir

```
<dependency>
    <groupId>io.micrometer</groupId>
    <artifactId>micrometer-registry-prometheus</artifactId>
</dependency>
```

```
management:
  health:
    circuitbreakers:
      enabled: true
  endpoints:
    web:
      exposure:
        include: health,info,prometheus
  metrics:
    export:
      prometheus:
        enabled: true
```

Métricas de negocio

```
@PostMapping
public ResponseEntity<PaymentDto> save(
    @RequestBody @NotNull(message = "Input must not be NULL") @Valid
final PaymentDto paymentDto) {
    log.info("**** PaymentDto, resource; save payment *");
    return ResponseEntity.ok(this.paymentService.save(paymentDto));
}
```

@Slf4j: Genera automáticamente un logger SLF4J
log.info(): Escribe logs que van a stdout/stderr del contenedor

```
@RestController
@RequestMapping("/api/payments")
@Slf4j // ← Anotación de Lombok para logging
@RequiredArgsConstructor
public class PaymentResource {

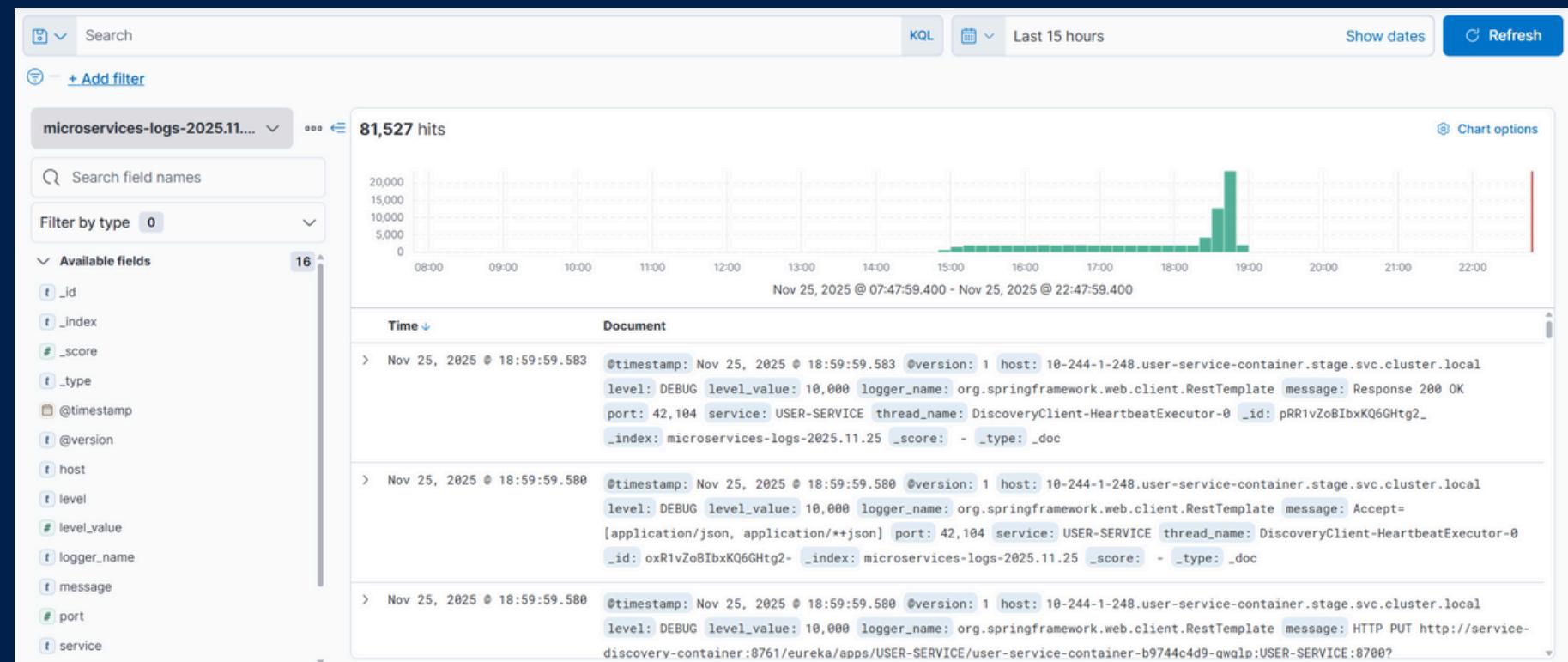
    @GetMapping
    public ResponseEntity<DtoCollectionResponse<PaymentDto>> findAll() {
        log.info("**** PaymentDto List, controller; fetch all payments *");
        return ResponseEntity.ok(new DtoCollectionResponse<>
            (this.paymentService.findAll()));
    }
}
```

Monitoreo

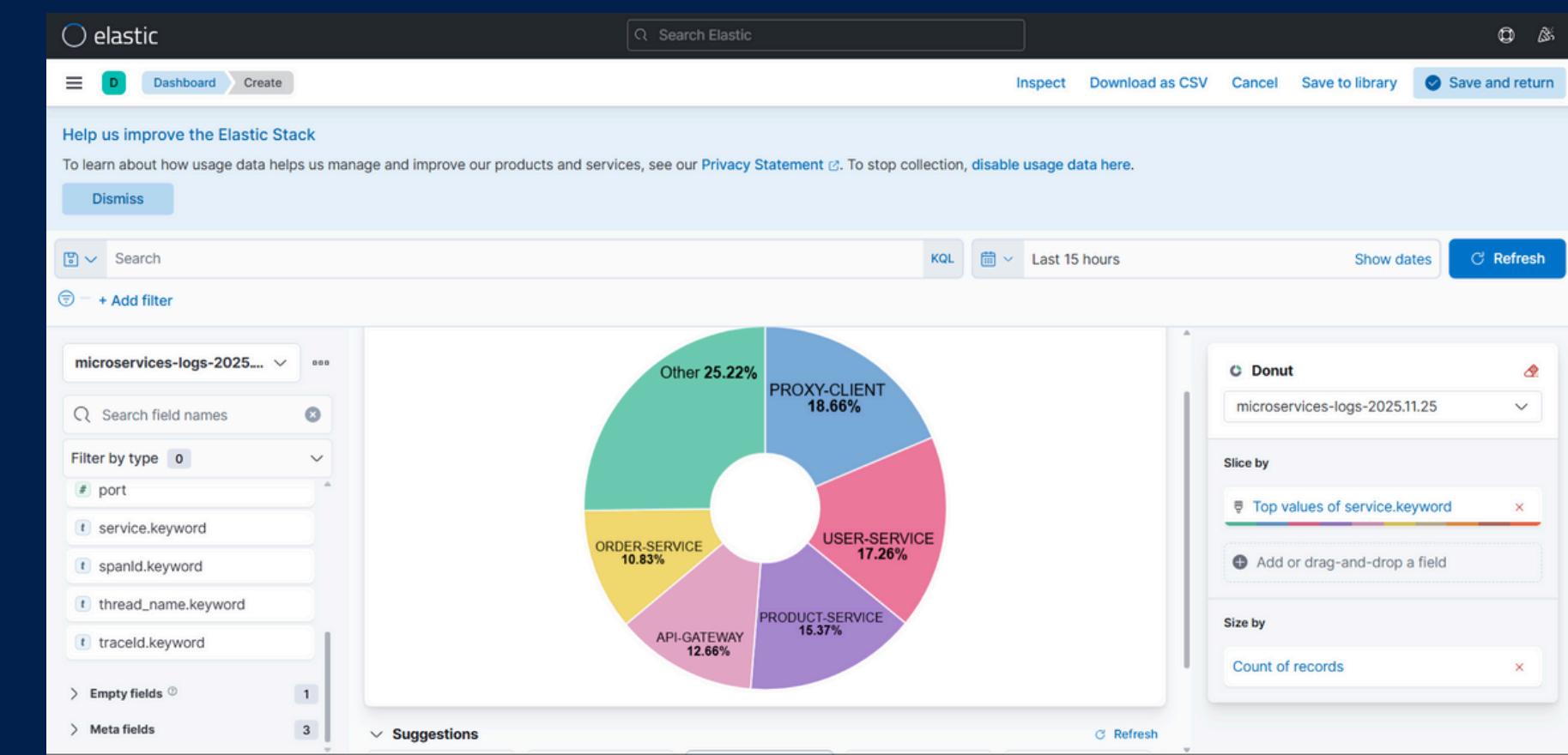
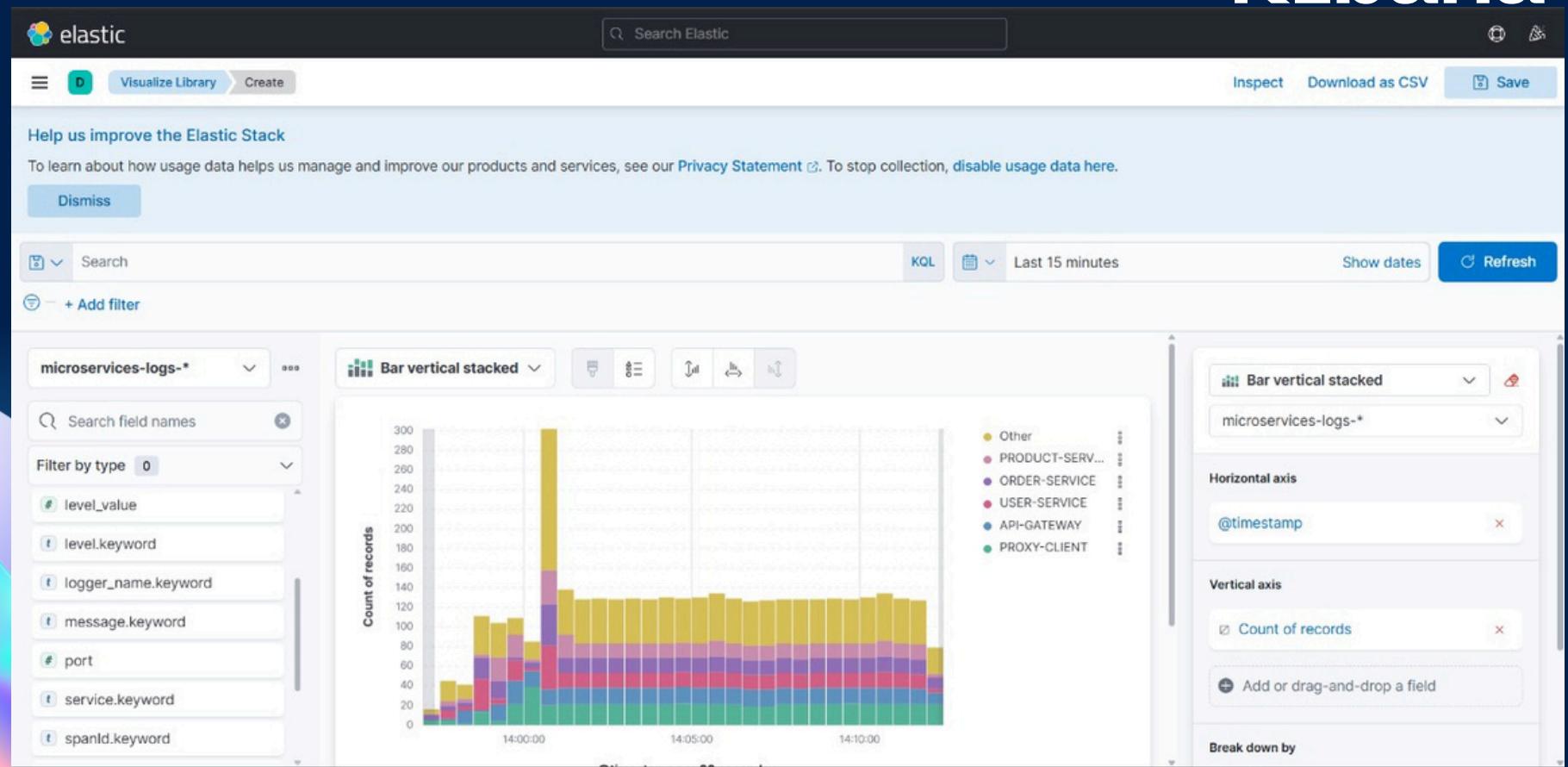
Kibana Logs

Prometheus Health Checks

The screenshot shows the Prometheus Targets page. At the top, there's a search bar and a filter bar for endpoints or labels. Below that is a table with columns: Endpoint, State, Labels, Last Scrape, Scrape Duration, and Error. One row is shown for the endpoint `http://10.244.1.248:8700/user-service/actuator/prometheus`, which is labeled "UP". The "Labels" column contains several key-value pairs: `container="user-service-container"`, `endpoint="http"`, `instance="10.244.1.248:8700"`, `job="user-service-container"`, `namespace="stage"`, `pod="user-service-container-b9744c4d9-gwglp"`, and `service="user-service-container"`.

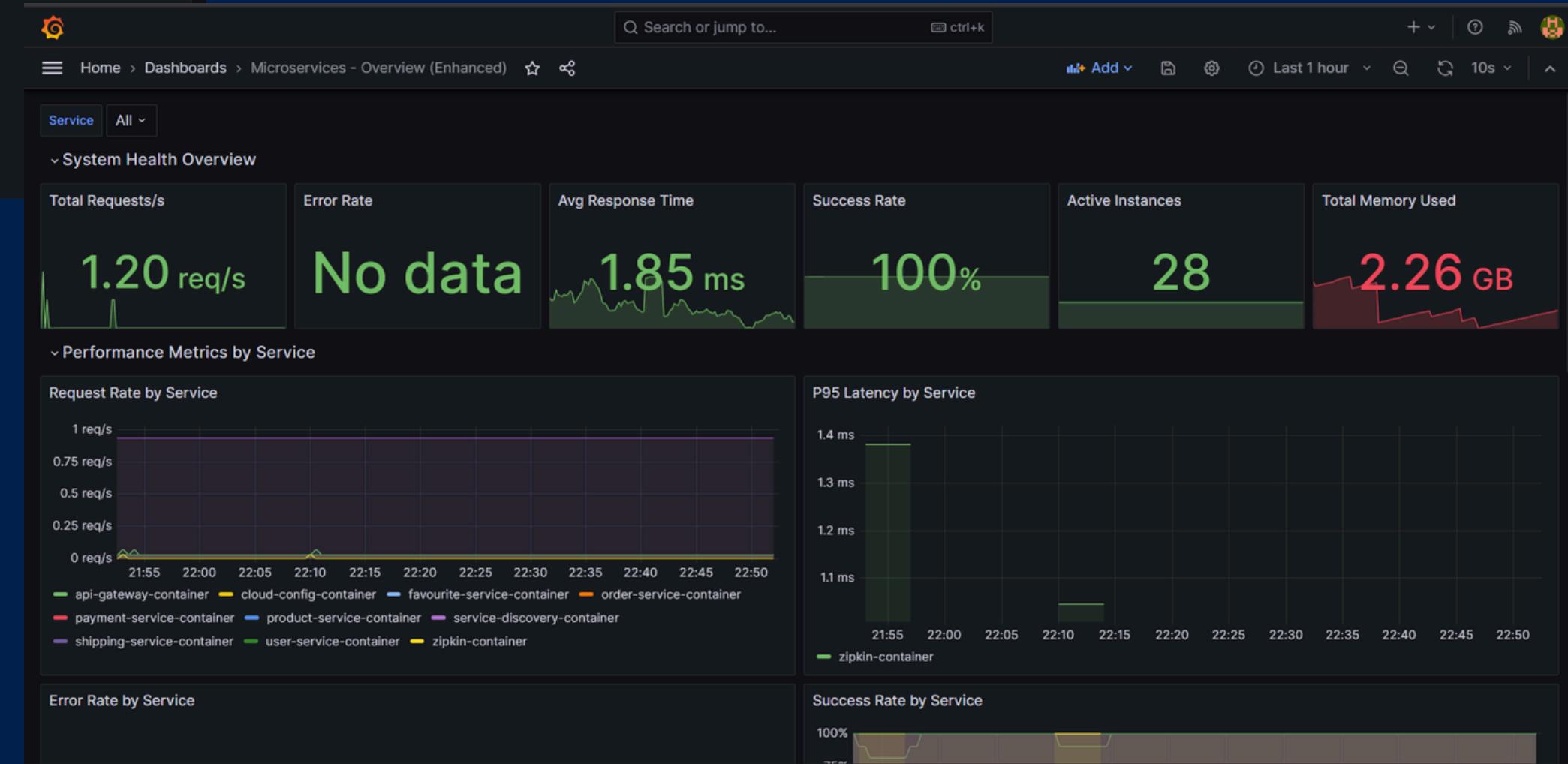
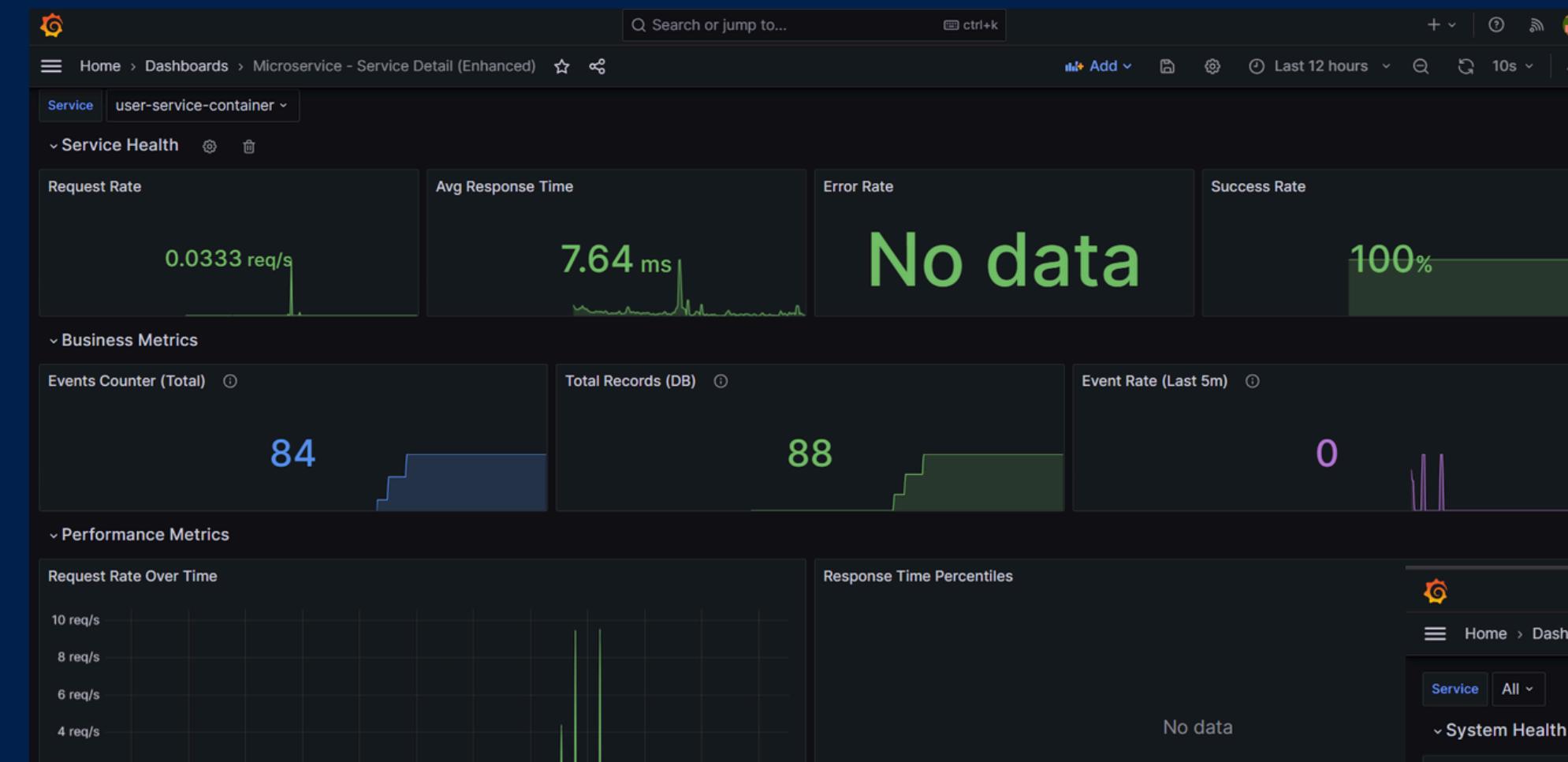


Kibana Dashboards



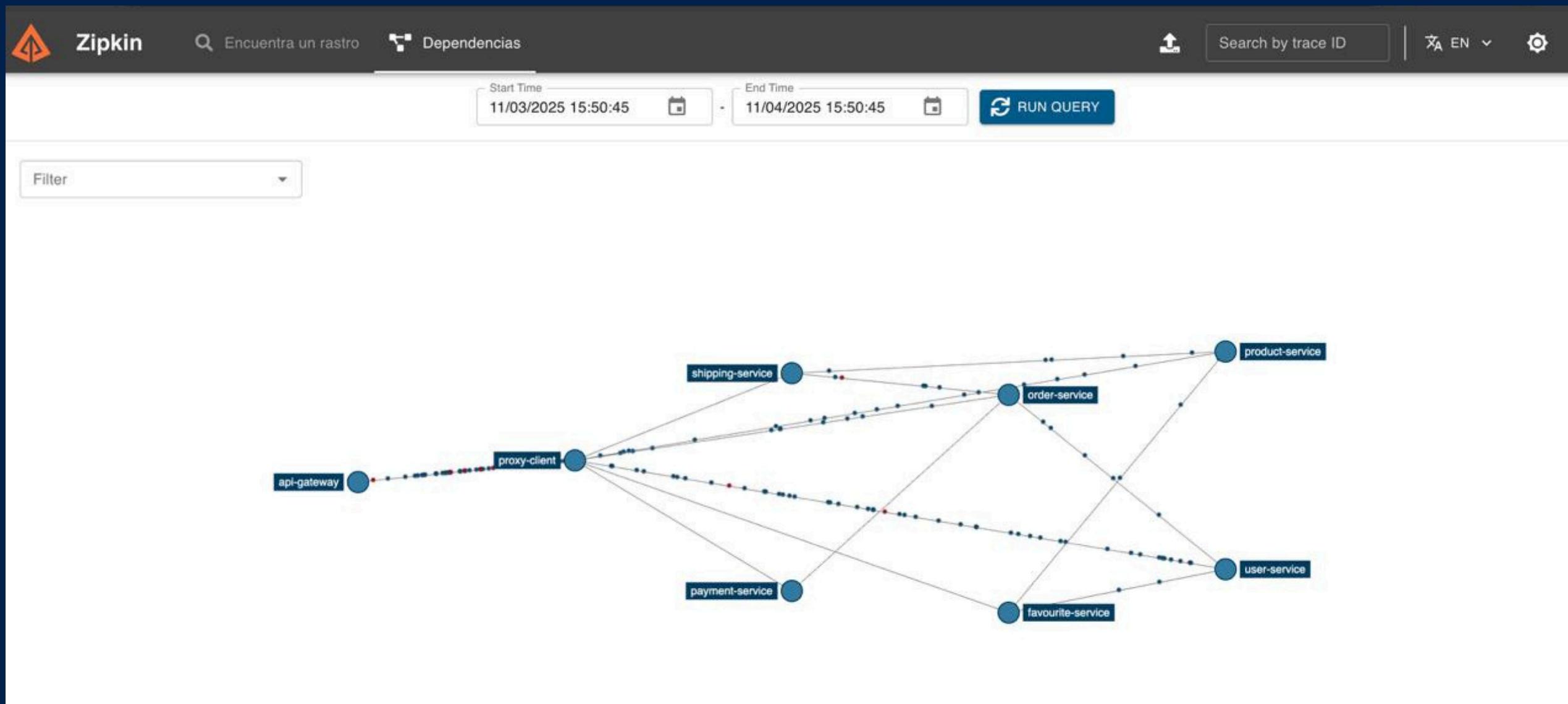
Monitoreo

Grafana Dashboards



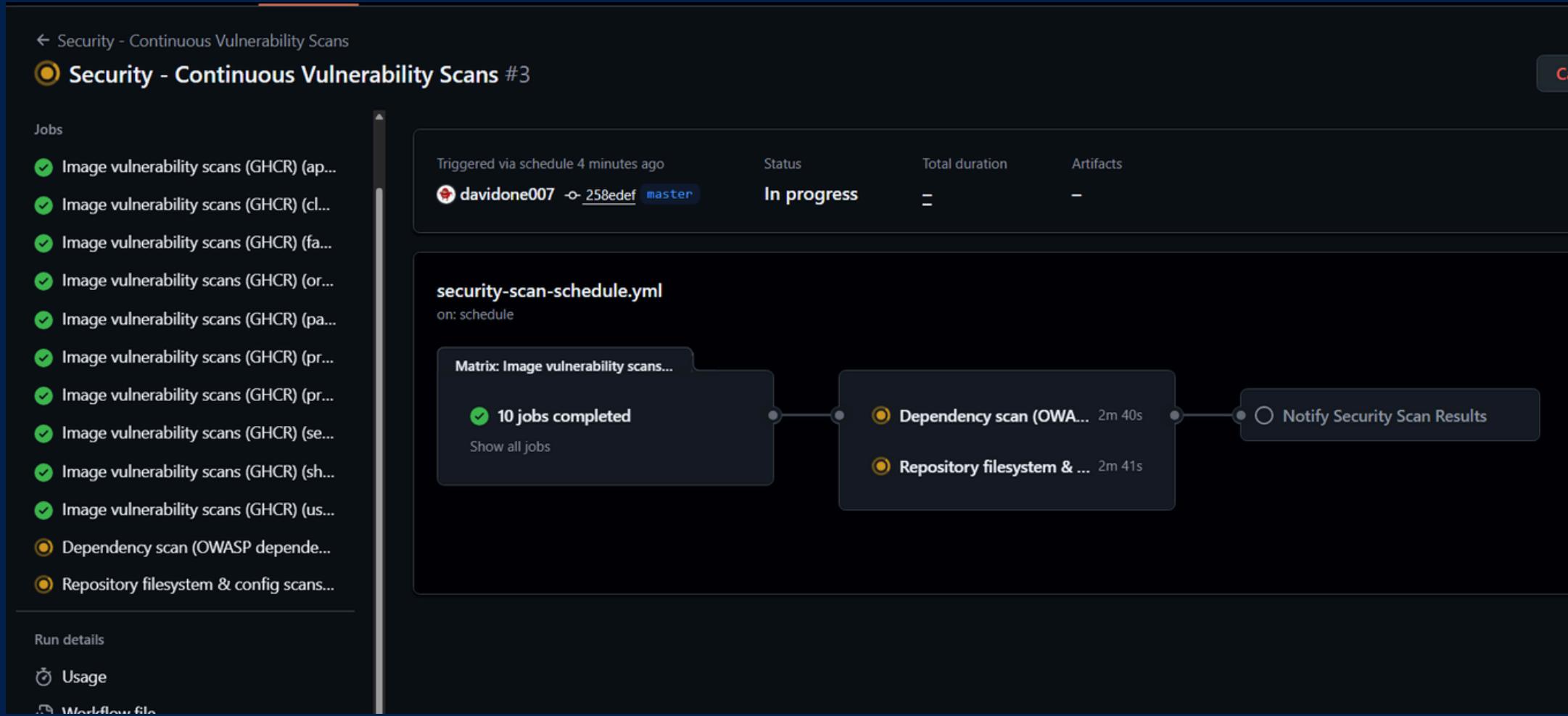
Monitoreo

Zipkin



SEGURIDAD

Escaneo continuo de vulnerabilidades



- Se ejecuta diariamente a las 02:00 UTC (21:00 COL) y también manualmente.
- Escanea imágenes Docker en GHCR (tags dev y latest) con Trivy.
- Ejecuta OWASP Dependency-Check para revisar vulnerabilidades en dependencias Maven.
- Realiza escaneo del repositorio:
- Vulnerabilidades en archivos (Trivy FS)
- Analiza en Dockerfile, YAML, Helm, Terraform (Trivy Config)
- Genera y guarda reportes completos como artefactos.

Gestión segura de secretos

```
❯ az keyvault secret show --vault-name ecom-kv-stg-glblzh --name jwt-secret --query value -o tsv  
secret123
```

```
❯ kubectl exec -n stage proxy-client-container-6bb799cc66-885r4 -- cat /mnt/secrets-store/jwt-secret  
Defaulted container "proxy-client-container" out of: proxy-client-container, wait-for-dependencies (init)  
secret123%
```

Key Vault centralizado:
terraform/modules/keyvault/

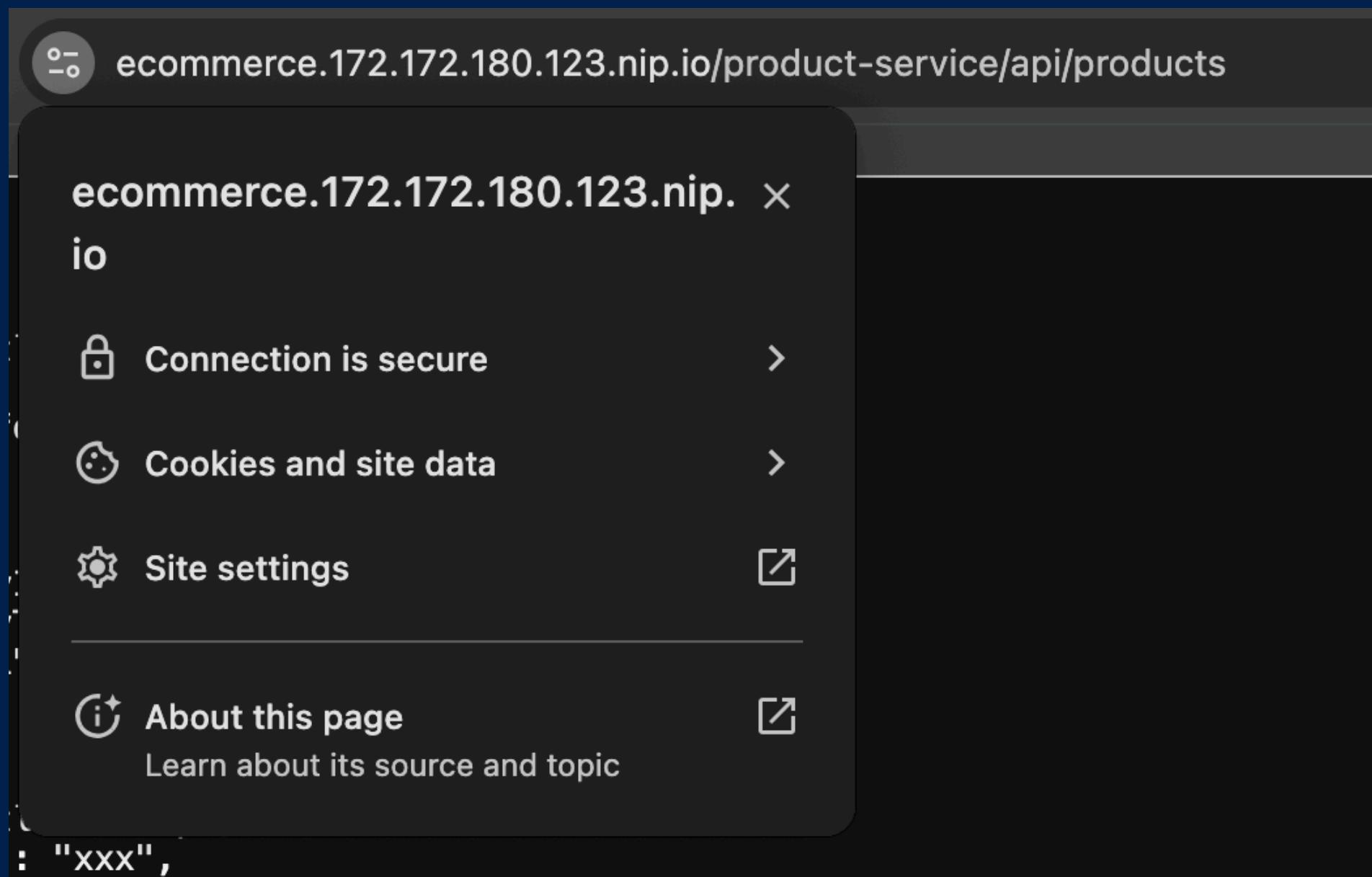
RBAC para acceso a recursos

```
Apply complete! Resources: 0 added, 1 changed, 0 destroyed.  
● > kubectl auth can-i get pods --as=system:serviceaccount:prod:dev-user -n prod  
yes  
⊗ > kubectl auth can-i delete pods --as=system:serviceaccount:prod:dev-user -n prod  
no
```

- Se crea una ServiceAccount (dev-user) que actúa como usuario dentro del namespace.
- Se define un Role (dev-reader) con permisos solo de lectura: get, list, watch.
- El Role permite leer recursos como: Pods, Services, Deployments, Ingresses, ConfigMaps y Secrets.
- Se crea un RoleBinding que vincula la ServiceAccount con el Role en el mismo namespace.
- El usuario resultante solo puede visualizar recursos del namespace, sin crear, modificar o borrar nada.
- Útil para: desarrolladores, QA o monitoreo con permisos limitados.

```
# 1. Creamos una ServiceAccount simulando un usuario desarrollador  
You, 2 days ago | 1 author (You)  
resource "kubernetes_service_account" "dev_user" {  
  You, 2 days ago | 1 author (You)  
  metadata {  
    name      = "dev-user"  
    namespace = var.namespace  
  }  
}  
  
# 2. Creamos un Role con permisos de lectura  
You, 2 days ago | 1 author (You)  
resource "kubernetes_role" "dev_reader" {  
  You, 2 days ago | 1 author (You)  
  metadata {  
    name      = "dev-reader"  
    namespace = var.namespace  
  }  
  rule {  
    api_groups = [ "", "apps", "networking.k8s.io" ]  
    resources = [ "pods", "services", "deployments", "ingresses", "configmaps", "  
    verbs     = [ "get", "list", "watch" ]  
  }  
}
```

TLS para servicios públicos



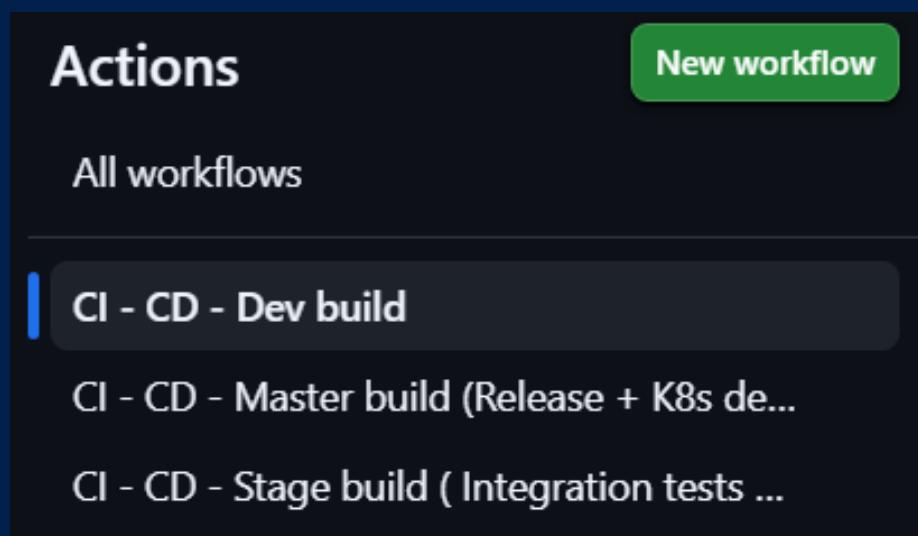
- Cert-manager gestiona emisión/renovación de TLS usando Let's Encrypt (ACME HTTP-01).
- El Ingress tiene la anotación `cert-manager.io/cluster-issuer: letsencrypt-prod` y un `secretName` para el TLS; cert-manager crea la Certificate y el Secret automáticamente y renueva antes de expirar.

CHANGE MANAGEMENT RELEASE NOTES



Change management

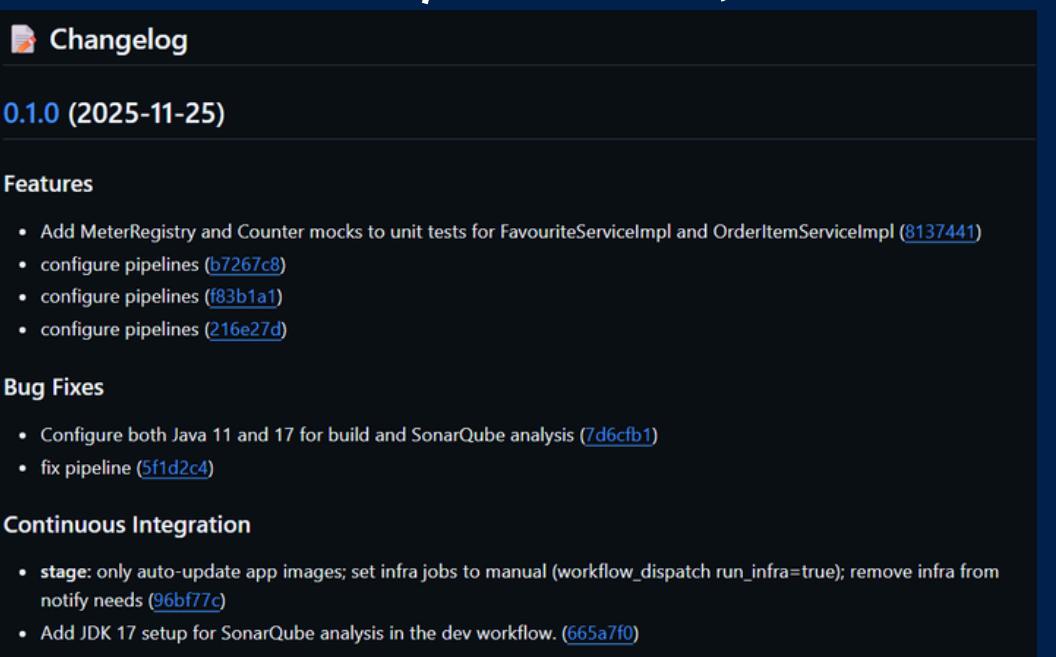
CI/CD Automático: GitHub Actions ejecuta tests en cada etapa antes de permitir merge



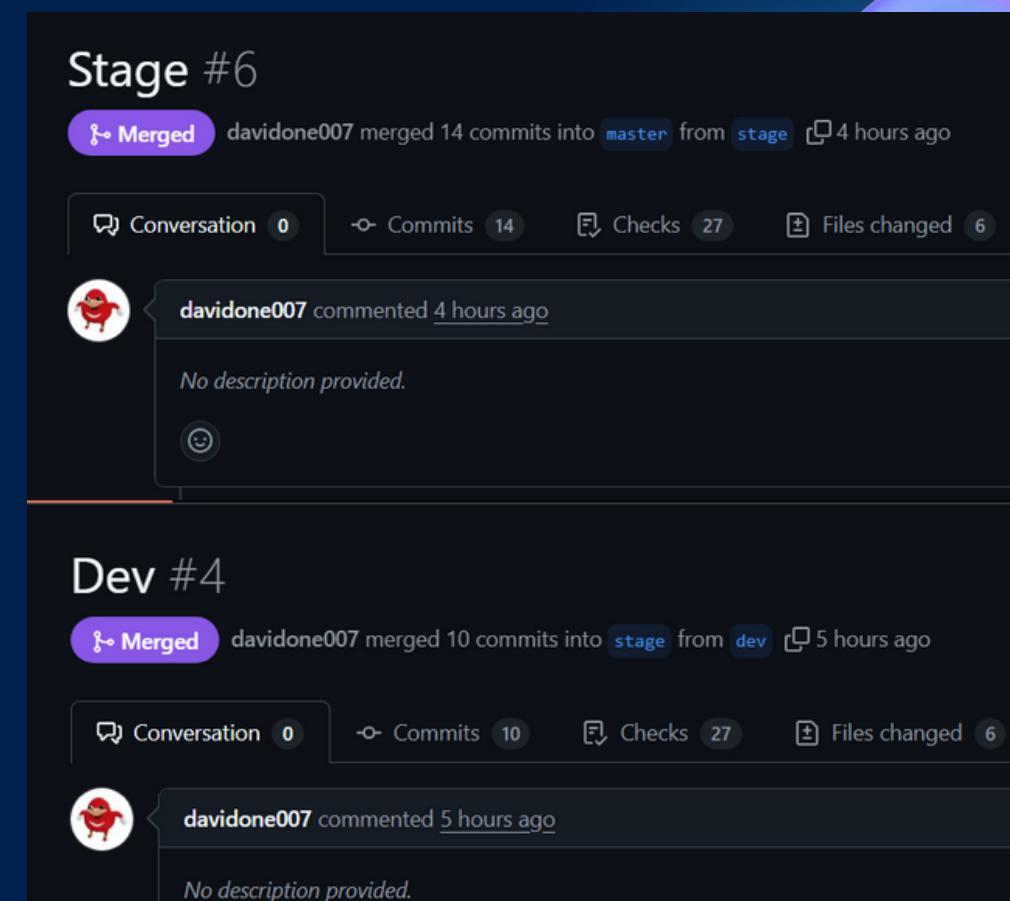
Release Notes: Documenta features, bugs corregidos e instrucciones para cada release



Changelog Documentado: Se registra qué se cambió en cada versión (v0.1.0, stage-0.1.0, dev-0.1.0)



Merges Controlados: Cada cambio se mergea entre branches (dev → stage → master)



Release Notes

3 hours ago

Release v0.1.0 Latest

Release v0.1.0

Changelog

Stage Release stage-0.1.0

Changelog

Dev Release dev-0.1.0

Changelog

0.1.0 (2025-11-25)

Features

- Add MeterRegistry and Counter mocks to unit tests for FavouriteServiceImpl and OrderItemServiceImpl ([8137441](#))
- configure pipelines ([b7267c8](#))
- configure pipelines ([f83b1a1](#))
- configure pipelines ([216e27d](#))

Bug Fixes

- Configure both Java 11 and 17 for build and SonarQube analysis ([7d6cfb1](#))
- fix pipeline ([5f1d2c4](#))

Continuous Integration

- stage: only auto-update app images; set infra jobs to manual (workflow_dispatch run_infra=true); remove infra from notify needs ([96bf77c](#))

Las Release Notes registran qué cambios, features, bugs corregidos e instrucciones de instalación incluye cada versión.

Artefactos Generados

- **Imágenes Docker con tag de versión**
- **JAR files como artifacts**
- **Changelog documentado**
- **Pipeline CI/CD validado**

Plan de Rollback

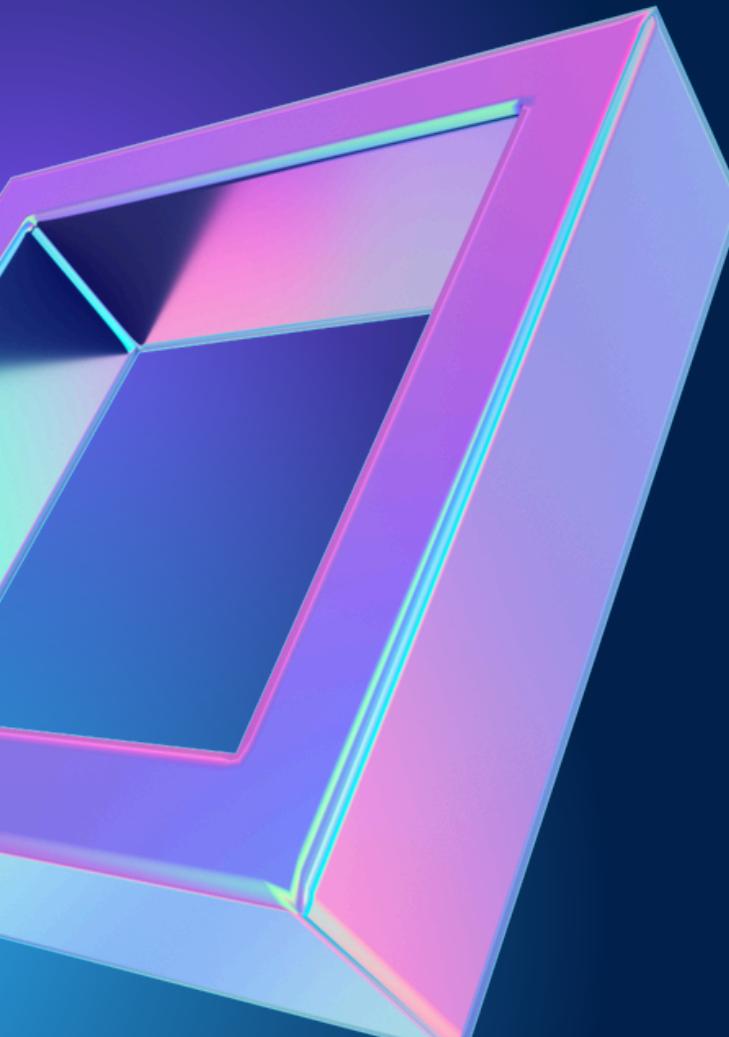
Plan de Rollback (Pasos)

- 1. Identificar el problema → Monitoreo detecta falla**
- 2. Decidir rollback → Si es crítico, revertir cambios**
- 3. Cambiar imagen Docker → Actualizar a versión anterior (v0.0.2 en lugar de v0.0.3)**
- 4. Redeploy → Aplicar cambios en Kubernetes**
- 5. Verificar → Confirmar que producción funciona**



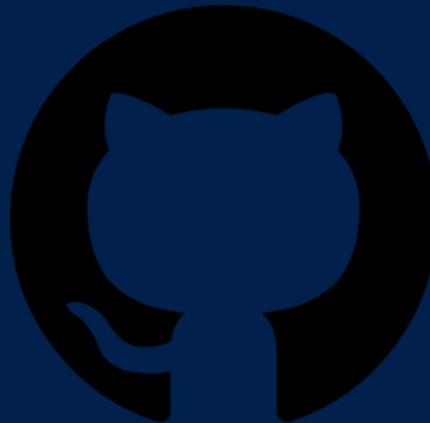
LINK DEPLOYMENT

<https://ecommerce.172.172.180.123.nip.io>



andrescabezas26

davidone007



Gracias