



# **Centro de Enseñanza Técnica Industrial**

## **Desarrollo de Software**

### **Operación Continua**

**Jesús Alberto Aréchiga Carrillo**

**22310439      6N**

**Profesor**

**José Francisco Pérez Reyes**

**Mayo 2025**

**Guadalajara, Jalisco**

# Implementación de un Sistema de Monitoreo con Prometheus

Para asegurar la operación continua, es necesario contar con una herramienta de monitoreo que recoja métricas en tiempo real, alerte sobre anomalías y facilite la visualización gráfica del comportamiento de cada microservicio.

## ¿Por qué Prometheus?

Prometheus es un sistema de monitoreo y almacenamiento de series temporales diseñado para entornos de microservicios.

- Recopilación basada en “pull”, donde el servidor Prometheus interroga periódicamente a los microservicios para recopilar métricas expuestas en formato estándar.
- Lenguaje de consultas PromQL, que permite definir consultas y reglas complejas para calcular indicadores y disparar alertas.
- Alertmanager integrado, que gestiona notificaciones cuando las métricas superan umbrales definidos.
- Ecosistema amplio, con exportadores para Kubernetes, Docker, Linux, bases de datos, y bibliotecas clientes para instrumentar cada lenguaje de programación.

## Monitoreo en Storix

La arquitectura de monitoreo con Prometheus se estructura en los siguientes componentes:

### 1. Servidor Prometheus

- Ejecuta la recolección de métricas desde cada microservicio y desde componentes de infraestructura.
- Almacena series temporales en su base de datos interna de alto rendimiento.
- Evalúa las reglas de alerta definidas mediante PromQL y envía alertas a Alertmanager.

### 2. Exportadores

- `node_exporter`: Recolecta métricas del sistema operativo de cada nodo (CPU, memoria, disco, red).

- Advisor: Extrae métricas de contenedores Docker/Kubernetes, como uso de CPU y memoria por contenedor.
- Instrumentación interna: Cada microservicio en Storix expone métricas personalizadas (número de solicitudes, latencia, errores) en un endpoint /metrics mediante bibliotecas oficiales de Prometheus.

### **3. Alertmanager**

- Recibe las notificaciones de alerta generadas por el servidor Prometheus tras evaluar las “alert rules”.
- Gestiona la agrupación de alertas, los silenciamientos y reenvíos a canales externos.