

Rubrica de Evaluación 2 (Técnicas de Ingeniería de Software)

1: La aplicación web debe estar funcionando completamente en Kubernetes (20%)

La aplicación web (frontend y backend), incluyendo todas las Funcionalidades definidas (Épica1 a Épica7), está completamente funcional y accesible desde Kubernetes. La comunicación entre frontend y backend funciona sin problemas, permitiendo un uso fluido desde el navegador web. *IMPORTANTE: No se acepta el uso de port-forward para redirigir el puerto local de la computadora hacia un Pod (o Service) dentro del cluster de Kubernetes (minikube).*

2. Correcta implementación de microservicios, y comunicación entre ellos (20%)

Cada uno de los microservicios (M1 a M7) implementa correctamente cada una de las Funcionalidades (Épica1 a Épica7) siguiendo una arquitectura de capas y está conectado exclusivamente a su propia base de datos relacional según corresponda, gestionando sus datos de forma independiente. Los microservicios se comunican entre sí exclusivamente mediante peticiones HTTP utilizando *RestTemplate* (u otro con la debida justificación), sin incluir direcciones IP ni puertos estáticos en las llamadas. Los puertos de los microservicios que correspondan se asignan dinámicamente (*server.port=0*), y la resolución de nombres se realiza correctamente a través del *Service Discovery (Eureka Server)*.

3. Implementación de patrones de microservicios (30%)

Se han implementado correctamente los patrones de microservicios solicitados: *ConfigServer*, *Service Discovery (Eureka Server)*, *API Gateway*, junto a los microservicios (M1 a M7). Cada patrón funciona según lo esperado, con *ConfigServer* gestionando configuraciones centralizadas, *Eureka Server* manejando la detección dinámica de servicios, y *API Gateway* sirviendo como único punto de acceso al backend. La implementación de estos patrones debe dejar la aplicación web correctamente funcionando (Épica1 a Épica7).

4. Uso correcto de los objetos de Kubernetes para el despliegue de la aplicación web (30%)

Se han utilizado correctamente todos los objetos de Kubernetes descritos en clase de cátedra para garantizar el despliegue correcto de la aplicación web (frontend y backend): *Pods*, *Deployments*, *Services*, *ConfigMaps*, y *Secrets*. Cada microservicio está empaquetado en un contenedor Docker y desplegado en Kubernetes con acceso a su propia base de datos relacional. El despliegue final debe dejar la aplicación correctamente funcionando (Épica1 a Épica7).

IMPORTANTE:

- En el desarrollo y despliegue de la aplicación, se deben tomar en consideración además las indicaciones dadas en el enunciado de la Evaluación 2.