

Datos del estudiante

Nombre y apellidos	Nombre y apellidos del estudiante
Fecha de entrega	00/00/0000

API REST en Spring Boot con Integración Continua y Despliegue en la nube (GitHub Actions + AWS Elastic Beanstalk)

Objetivos de la actividad

El objetivo de esta práctica es que el alumnado implemente un flujo real de Integración Continua (CI) y Despliegue Continuo (CD) para una aplicación backend desarrollada en Java con Spring Boot, utilizando herramientas profesionales del entorno DevOps.

Al finalizar la práctica, el alumno será capaz de:

- Desarrollar una API REST mínima con Spring Boot.
- Gestionar correctamente el control de versiones con Git.
- Automatizar la construcción y validación del proyecto mediante GitHub Actions.
- Desplegar automáticamente una aplicación Java en la nube con AWS Elastic Beanstalk.
- Comprender el impacto del uso de recursos cloud y aplicar buenas prácticas de ahorro de costes.

Descripción general

Se desarrollará una API REST sencilla cuyo objetivo no es la funcionalidad, sino validar el proceso completo de automatización desde el código hasta producción.

Cada vez que se realice un push al repositorio remoto:

1. GitHub Actions ejecutará el pipeline de integración continua.
2. El proyecto se compilará con Maven.
3. Se generará un artefacto ejecutable (JAR).

4. La aplicación se desplegará automáticamente en AWS Elastic Beanstalk.

Se permiten despliegues manuales una vez configurado el pipeline debido al laboratorio de AWS

3. Requisitos técnicos

- Java 17
- Spring Boot 3.x
- Maven (con Maven Wrapper)
- Git y GitHub
- GitHub Actions
- AWS Elastic Beanstalk (Java SE)

Pasos a seguir:

Fase 1 – Desarrollo de la API Spring Boot

La aplicación debe incluir, como mínimo, los siguientes endpoints:

- GET /

Devuelve un mensaje de texto indicando el fin de la práctica.

- GET /api/estado

Devuelve un JSON con información básica del servicio (estado, nombre y mensaje).

La aplicación debe ejecutarse correctamente en local y generar un JAR ejecutable.

Fase 2 – Control de versiones

El proyecto debe gestionarse mediante Git, aplicando buenas prácticas:

- Uso de .gitignore para excluir archivos generados.
- Commits descriptivos y coherentes.
- Uso de la rama principal (main).

Fase 3 – Integración Continua con GitHub Actions (CI)

Se debe configurar un pipeline de integración continua que:

- Se ejecute automáticamente en cada push a la rama main.
- Compile el proyecto usando el Maven Wrapper.
- Ejecute la fase de tests (mvn test).

El pipeline debe finalizar en estado Success antes de continuar con el despliegue.

Fase 4 – Despliegue Continuo con AWS Elastic Beanstalk (CD)

Se debe configurar un entorno en AWS Elastic Beanstalk que permita:

- Desplegar una aplicación Java SE basada en un JAR ejecutable.
- Utilizar un paquete de despliegue limpio (deploy.zip) que contenga:
 - app.jar
 - Procfile

El despliegue debe realizarse automáticamente desde GitHub Actions tras cada push.

Flujo final del sistema

El flujo completo del proyecto debe ser el siguiente:

git commit



git push



GitHub Actions (CI)



Compilación y tests



Empaquetado del artefacto



AWS Elastic Beanstalk (CD)



Aplicación actualizada en producción

6. Entrega de la práctica

El alumno deberá entregar:

1. URL del repositorio GitHub con el proyecto completo.
2. Captura del pipeline en estado Success en GitHub Actions.
3. URL pública del entorno de Elastic Beanstalk donde se muestre:
 - el endpoint raíz (/)
 - el endpoint /api/estado
4. Breve explicación escrita (máx. 1 página) donde se describa:
 - el funcionamiento del pipeline,
 - el flujo CI/CD,
 - y las decisiones tomadas.

Extensión y formato

Formato: Documento PDF.

Extensión máxima: ocho páginas.

Formato del texto: fuente Calibri, tamaño 11, interlineado 1,5.

Elementos obligatorios: esquema de arquitectura (puede ser Draw.io, Lucidchart, etc.), capturas de consola o comandos (reales o simulados), comentarios explicativos, justificación de diseño, código IaC (opcional como anexo).

Criterios de evaluación

Criterio	Peso
API Spring Boot funcional	10 %
Uso correcto de Git y commits	20 %
Pipeline de CI correctamente configurado	30 %
Despliegue automático en Elastic Beanstalk	30 %
Explicación y comprensión del flujo CI/CD	10 %
Total	100%