

Desafío - Sistema de gestión de tareas

En este desafío pondrás a prueba tus conocimientos sobre integración continua y desarrollo de una API backend utilizando tecnologías modernas de desarrollo de software. Para su desarrollo, podrás apoyarte del recurso “Apoyo desafío - Sistema de gestión de tareas”, disponible en plataforma.

Descripción

La empresa ficticia CodeCraft está desarrollando una API básica para gestionar tareas en Node.js. Como parte del equipo backend, se te ha asignado la tarea de configurar un pipeline de integración continua que permita:

- Gestionar el código fuente con Git y un repositorio remoto en GitHub.
- Utilizar Docker para contenerizar la aplicación y garantizar la portabilidad.
- Configurar un pipeline en Jenkins que automatice el proceso de integración continua.

Requerimientos

1. Gestión del Repositorio Git (2 puntos)

- Inicializa un repositorio local y realiza un commit inicial con un archivo README.md.
- Crea y conecta un repositorio remoto en GitHub.

2. Configuración de la API con Docker (3 puntos)

- Escribe un **Dockerfile** que permita construir una imagen para ejecutar la API.
- La API debe responder en el puerto 3000 con rutas **/tasks** y **/tasks/:id**.

3. Pipeline de Jenkins (3 puntos)

- Configura Jenkins para clonar el repositorio desde GitHub.
- Define un pipeline que:
 - Instale dependencias (**npm install**).
 - Ejecute las pruebas automatizadas (**npm test**).
 - Construya una imagen Docker a partir del **Dockerfile**.

4. Reporte y Documentación (2 puntos)

- Genera un reporte con los resultados de las pruebas y el estado del pipeline.
- Documenta en un archivo **REPORT.md** los pasos realizados y cualquier problema encontrado.

¡Mucho éxito!



Consideraciones y recomendaciones

- Considera que los entregables son:
 - Código del proyecto (subido al repositorio de GitHub).
 - Capturas de pantalla del pipeline configurado en Jenkins y los reportes generados.
 - Archivo REPORT.md con la explicación de las configuraciones realizadas.
- El documento apoyo desafío contiene los siguientes archivos:
 - app.js: API básica con rutas /tasks y /tasks/:id.
 - tests/app.test.js: Pruebas unitarias para verificar la funcionalidad de la API.
 - package.json: Configuración del proyecto Node.js con dependencias necesarias.
 - Dockerfile: Archivo para contenerizar la API.
 - README.md: Descripción del proyecto y las instrucciones básicas.