

Challenge Sr Fullstack (Microservicios)

El objetivo de este desafío es evolucionar el sistema actual (catálogo e inventario) hacia un modelo event-driven, resolviendo primero los problemas estructurales, y exponer ese flujo a un frontend simple en React.

Se evalúa:

- Capacidad de diagnóstico
- Criterio técnico
- Diseño orientado a eventos
- Manejo de asincronía

A continuación, se detallan las tareas propuestas.

- 1) Descargar el siguiente repositorio y publicarlo en una cuenta propia:

<https://github.com/hsn656/nestjs-ecommerce>

El sistema representa un backend inicial de e-commerce (catálogo, inventario, usuarios) construido como un monolito en NestJS y PostgreSQL.

Importante

Es un repositorio público elegido para el desafío, el autor no pertenece a la empresa. No forkear.

La aplicación no está lista para producción. Contiene errores de diseño, arquitectura e implementación. Parte del desafío es identificar, corregir y justificar esos problemas para poder avanzar.

- 2) Diagnóstico inicial

Identificar problemas actuales del repositorio y aplicar los cambios mínimos necesarios para poder ejecutar la aplicación y tener una base razonable para evolucionar.

No se espera un refactor total, sino **decisiones justificadas**.

- 3) Alcance Backend

- Analizar el diseño actual y realizar los cambios mínimos necesarios para poder trabajar sobre él.
- Identificar **puntos naturales del dominio** donde tenga sentido emitir eventos.

- Diseñar e implementar **al menos 2 eventos de dominio relevantes**.
- Implementar **consumidores desacoplados** para esos eventos.
- Evitar comunicación directa entre módulos cuando no sea necesaria.

4) Alcance Frontend (React)

Crear una aplicación React simple que:

- Consuma información del backend
- Refleje cambios derivados de flujos asincrónicos (eventos)
- Permita validar que el sistema funciona de punta a punta

Aclaración: No se evalúa UI ni diseño visual.

5) Deploy

El sistema debe estar **deployado y accesible**:

- Backend
- Frontend
- Base de datos (puede ser gestionada)

El candidato puede elegir Cloud provider, infraestructura y herramientas. No se evalúa la plataforma, sino el criterio y la capacidad de llegar a producción.

6) Entregables

- Enlace del repositorio con la solución
- Documentación explicando:
 - Problemas detectados en el diseño original
 - Qué eventos se implementaron y por qué
 - Decisiones técnicas relevantes
 - Cómo levantar el proyecto
- URLs públicas de acceso