

# Backend · Docker · CI/CD

## 1) ¿Cómo se relacionan frontend, backend y base de datos dentro de una arquitectura web moderna?

Explica el rol de cada uno y cómo se comunican entre sí.

El frontend es lo que ve y usa el usuario, se trata de la parte de interacción visual entre la app y el usuario.

El backend es el que se encarga de la lógica de negocio y las reglas de la app.

La base de datos guarda la información.

El frontend se comunica con el backend y el backend es el único que accede a la base de datos.

---

## 2) ¿Qué ventajas ofrece una API REST frente a otros modelos de comunicación entre sistemas?

Menciona al menos dos ventajas desde el punto de vista de escalabilidad o mantenimiento.

Permite que diferentes aplicaciones se comuniquen de forma sencilla y ordenada. Además, facilita el mantenimiento y hace que el sistema pueda crecer sin tener que rehacerlo todo.

---

## 3) ¿Qué implica diseñar un backend desacoplado del frontend?

Explica por qué es importante en proyectos grandes o a largo plazo.

Significa que el back y el front no dependen directamente uno del otro.

---

## 4) ¿Qué diferencias conceptuales existen entre contenedores Docker y máquinas virtuales?

Indica al menos dos diferencias claras a nivel de recursos o arquitectura.

Los contenedores usan menos recursos y se apoyan en el sistema operativo del host.

Las máquinas virtuales son más pesadas porque cada una tiene su propio sistema operativo completo.

---

**5 ¿Qué significa que una aplicación sea *portable* y cómo ayuda Docker a conseguirlo?**

Explica por qué esto es clave en entornos profesionales. Que la aplicación puede ejecutarse en distintos entornos sin cambios.

Docker lo consigue empaquetando todo lo necesario

---

**6 ¿Qué es la integración continua (CI) y qué problemas evita en un equipo de desarrollo?**

Relaciona la respuesta con errores de código o coordinación.

Es un proceso automático que comprueba que el código funciona cada vez que se hacen cambios. Evita errores y fallos cuando se trabaja en equipo.

---

**7 ¿Qué es el despliegue continuo (CD) y qué requisitos debería cumplir un proyecto para poder usarlo con seguridad?**

Es el proceso de publicar cambios automáticamente en producción. Para hacerlo con seguridad el proyecto debe ser estable, tener tests y un buen control de versiones.

---

**8 ¿Por qué la automatización es un elemento clave en DevOps?**

Porque ahorra tiempo, reduce errores humanos y mejora la calidad del software. Además, permite que el equipo se centre en desarrollar en lugar de repetir tareas manuales.

---

**9 ¿Qué diferencias existen entre un error detectado en fase de desarrollo y uno detectado en producción?**

Explica las consecuencias técnicas y organizativas.

Un error en desarrollo se puede corregir fácilmente. Un error en producción afecta a usuarios reales, genera más presión y puede causar problemas graves en el sistema o en la empresa.

---

**10 ¿Por qué es importante separar la configuración del código en un backend?**

Da ejemplos conceptuales (entornos, seguridad, mantenimiento).

Porque la configuración puede cambiar según el entorno sin tocar el código. Esto mejora la seguridad, el mantenimiento y facilita trabajar con distintos entornos como desarrollo o producción.