



Video al que se hace referencia de [preguntas frecuentes en CQRS y Domain-Driven Design](#).

Posibles responsables para generar el identificador

MySqlVideoRepository

En este caso haríamos el insert sin especificar el identificador del vídeo. A posteriori sería cuando **le pediríamos a MySQL el último identificador generado**, o todos los datos del vídeo recién creado incluyendo el ID que se le ha asignado.

Contras

Delegamos la responsabilidad de gestionar los identificadores a un componente de infraestructura. Desde el momento en el que movemos esto a infraestructura, sería un aspecto más a tener en cuenta a la hora de cambiar este componente. Con lo cuál, dificultaría la transición de un adaptador a otro.

Representaciones de recursos inconsistentes en cliente. El cliente nos hace la petición sin el identificador, ya que es el backend quien genera el ID. Esto implica que el cliente permita que el ID sea nulo en ciertos momentos.

Complejidad adicional en testing. Desde el momento en el que la generación de identificadores no es determinista, no podemos comparar que todos los atributos de nuestros recursos sean iguales. Lo que deberemos hacer será comprobar que todos los atributos, a excepción del identificador, coinciden con los del recurso que le hemos pedido al sistema que cree.

VideoCreator (servicio / caso de uso)

Beneficios

Ahora ya no tenemos el primero de los contras anteriores. **Podemos cambiar la infraestructura manteniendo la lógica de gestión de identificadores.**

Contras

Seguimos teniendo los problemas que analizábamos antes en términos de inconsistencia en clientes, y complejidad de testing. No obstante, sí que es cierto que en este caso podríamos encapsular la lógica de generación de IDs en un servicio para poder hacer un doble de test y que éste fuera determinista. No obstante, estaríamos añadiendo una complejidad tanto al testing como al código de producción que nos podemos ahorrar.

Cliente (app, JS del frontend, etc.)

Beneficios

Mantenimiento y cambiabilidad: Podemos cambiar cualquier componente de infraestructura que ello no implicará tener que replicar la lógica de gestión de identificadores, ni tenerla duplicada entre todos los adaptadores.

Testing: Ya podemos comparar la entidad que le pedimos crear a nuestro sistema de forma íntegra (sin obviar el atributo de ID).

Integridad de representación de recursos en cliente: El cliente va a poder establecer el atributo del identificador como obligatorio (no null) ya que en todo momento va a disponer de éste.

¡Nos vemos en el siguiente vídeo!

• **Discuss:** ¿Qué beneficios y contras le ves a este enfoque? [Answer >](#)

• **Discuss:** ¿Tienes ejemplos de uso de este tipo de prácticas? [Answer >](#)

Discuss: ¿En qué punto del repositorio de ejemplo se puede ver esto implementado? [Answer >](#)

Next >

🏆 You earned 100/100pts + 🔄 Reset



Have a question about this step? Start a discussion to get feedback from others in this course

Start a discussion

