VERSION 6 de backup

SELECT r.id\_respuesta, r.texto, a.nombre\_autor, r.es\_validada

FROM respuestas r

INNER JOIN mensajes m ON r.mensaje\_id = m.id\_mensaje

INNER JOIN autores a ON a.id\_autor = m.autor\_id

WHERE r.es\_validada = TRUE AND a.es\_docente = FALSE;

|  |  |  |  |
| --- | --- | --- | --- |
| Id\_respuesta | texto | nombre\_autor | es\_validada |
| 142 | "pero ya lo solucione, borre el código del pom y lo volvi a escribir. No se que paso" | "feinez\_" | true |
| 337 | "Sisi, el ejercicio en lo que al código respecta se entiende perfecto y también hay un diagrama de secuencia subido, pero tenía esa duda con el diagrama de clases.  Ahora ya entendí, muchas gracias!" | "ivanabete" | true |
| 618 | "Consulta del final de esTim.  ¿Está bien pensar en una tabla ""juego\_adquirido"" para saber qué juegos pertenecen al usuario?  Porque puede que sea un poco redundante con la tabla ""compra\_juego"" que, aunque guarde las compras del juego en si, la realidad es que un juego comprado sería un juego adquirido (o así se podría interpretar), entonces se podrían obtener los juegos adquiridos del usuario de dicha tabla.  Sin embargo, me gusta la alternativa de tener aparte la tabla juego\_adquirido, creo que facilita la interpretación del modelo (y a lo mejor me ahorra algún join).  ¿Estaría bien pensarlo así?  Siguiendo esa lógica, a prestamo\_juego le puse una FK a juego\_adquirido (juego\_prestado\_id), así ya conoce al juego y al prestador." | "ivanabete" | true |
| 935 | "Buenas quiero sumar un poquito a lo que pregunto uri hace un rato.  Me marié un poco a la hora de diseñar los adapters que se usan para consumir api\_rest, porque hasta el momento, por lo general yo venia haciendo una firma por ejemplo \*\*consultarVeracidadPersona(Persona): bool\*\* en la cual asumía que se consultaba una api externa y esta te devolvía un json con datos y luego de ahí deducir si la persona es veraz o no.  El problema con esto es que al definir un valor de retorno estaría medio forzando a que la integracion sea sincrónica (A menos que use algún recurso tipo hilos y me quede esperando ahi)  En el caso del webhook como hay que definir un endpoint aparte entonces la firma pasaría a ser void \*\*consultarVeracidadPersona(Persona)\*\*  Entonces cambia completamente la firma del adapter según el tipo de integración  Ademas sigo medio dando vueltas con el tema de quien define si la respuesta es sincrónica o asincronica. si el que te expone la api o nosotros como clientes de la misma. Puesto que en varios finales solo dice algo como ""consulto tal cosa mediante una API REST"" y no hay mucho mas detalle que eso" | "gonzaloturri" | true |
| 940 | "Buenas quiero sumar un poquito a lo que pregunto uri hace un rato.  Me marié un poco a la hora de diseñar los adapters que se usan para consumir api\_rest, porque hasta el momento, por lo general yo venia haciendo una firma por ejemplo \*\*consultarVeracidadPersona(Persona): bool\*\* en la cual asumía que se consultaba una api externa y esta te devolvía un json con datos y luego de ahí deducir si la persona es veraz o no.  El problema con esto es que al definir un valor de retorno estaría medio forzando a que la integracion sea sincrónica (A menos que use algún recurso tipo hilos y me quede esperando ahi)  En el caso del webhook como hay que definir un endpoint aparte entonces la firma pasaría a ser void \*\*consultarVeracidadPersona(Persona)\*\*  Entonces cambia completamente la firma del adapter según el tipo de integración  Ademas sigo medio dando vueltas con el tema de quien define si la respuesta es sincrónica o asincronica. si el que te expone la api o nosotros como clientes de la misma. Puesto que en varios finales solo dice algo como ""consulto tal cosa mediante una API REST"" y no hay mucho mas detalle que eso" | "gonzaloturri" | true |
| 949 | "Ayluu buenísimo. Claro en mi abstracción TipoElemento solo tengo un simple string para normalizar la información.  Entonces estaría bien una justificación de diseño así? Por simplicidad tengo la lista de acciones en el Elemento. Si es un elemento que no es de TipoElemento ""Notebook"" es instanciado tendrá en null la lista de acciones y el método ejecutar acciones no tendría comportamiento, lo mismo sucedería si no se decidiera ejecutar acciones sobre un elemento ""Notebook"" particular." | "ivanerazun" | true |
| 1078 | "creo que hoy es entrega de tp y la prox en medrano" | "camibarros" | true |
| 1130 | "Ahhh verdad que estaba esa guía! Mil gracias" | "cypher.punk" | true |
| 1135 | "Buenas! Ya pregunté, es el aula 4 y está abierta" | "skalusti" | true |
| 1144 | "Claro, o sea que cada entrada en la tabla sería el esquema de precios con la fecha cuando cambió?" | "cypher.punk | true |
| 1242 | "Hola buenas tardes! En algunos finales aparece que se pueda tener trazabilidad de los estados por los que pasa una entidad... Quisiera preguntarles si lo que estaría bien sería modelarlo simplemente como un enum en el caso que no tenga comportamiento alguno y sean siempre los mismos como en el primer caso. O si conviene hacer una clase aparte que tenga un enum con el estado y otro con un local date que diga cuando entro en ese estado..." | "nazarenolopez" | true |
| 1255 | "Sii perfecto! Mil gracias!! En Microservicios entonces seria externo y se conectaría directo con el API Gateway?" | "vickitoria" | true |
| 1446 | "Hola Lucas, gracias, ahi se soluciono, si alguno le pasa lo mismo, en arquitectura seleccione quickstart, disculpen se me paso ese detalle" | "ferrami" | true |
| 1626 | "Sí perfecto, muchas gracias!!" | "ceci.24" | true |
| 1750 | "Hola Lucas! Estamos estableciendo la conexión solo una vez, por eso nos llamaba la atención el error. No sabemos si, quizás queda tomada la conexión del broker, pero siempre que probamos ejecutandolo finalizamos la conexión antes de empezar una nueva" | "vickitoria" | true |