

Watch: **Fallando miserablemente modelando Agregados** 🤖

Fallando miserablemente modelando Agregados 🤖

Como podéis ver en la propia web (podéis acceder desde este [enlace](#)), El Menú que estamos referenciando se compone de varios Tabs, dentro de los cuales hay múltiples Filtros, que a su vez tendrían una lista de Enlaces

Con esta base, y con idea de empezar “jugar” con el tema de los agregados la propuesta que tuvimos fue establecer un *Aggregate Menu* con su correspondiente *Aggregate Root Menu*, que además de detalles como el idioma o la última modificación, contendría los Tab, y dentro estos los Filter, que a su vez tendría los Link

 menuAggregate

Con este planteamiento, si queremos añadir un nuevo Link, tendremos que ir haciendo múltiples llamadas desde el *Aggregate Root* hasta llegar a *Filter*, donde haríamos el new

Y no sólo eso, sino que *Menu* será una clase mastodóntica donde tendremos métodos de Añadir-Editar-Eliminar Menús, Pestañas, Filtros y Links

¿Qué falló?

- ¿CRUD era suficiente?
 - Distintas representaciones del menú en distintos idiomas con reglas complejas
- ¿Se podría haber evitado niveles de encapsulación?
 - Validación no más de X elementos dependiente de otros

- No por evento (acción ya realizada)
- Si por query/caso de uso
- ¿Y entonces tendríamos N consultas?
 - ¡Read Model!

¿Alguna Duda?

Recordad que si tenéis dudas u os gustaría afianzar más todos estos conceptos podéis verlo con todo lujo de detalle con los cursos de [Arquitectura Hexagonal](#), [CQRS](#) y [Comunicación entre Microservicios](#)

Si tienes cualquier pregunta o sugerencia sobre este video nos encantaría que nos lo contaras ¡Abre una nueva discusión aquí abajo y escríbenos! 🙌🙌🙌

¡Nos vemos en la próxima lección! 🌐 Patrón Repository