

Aplicaciones Web Orientadas a Servicios

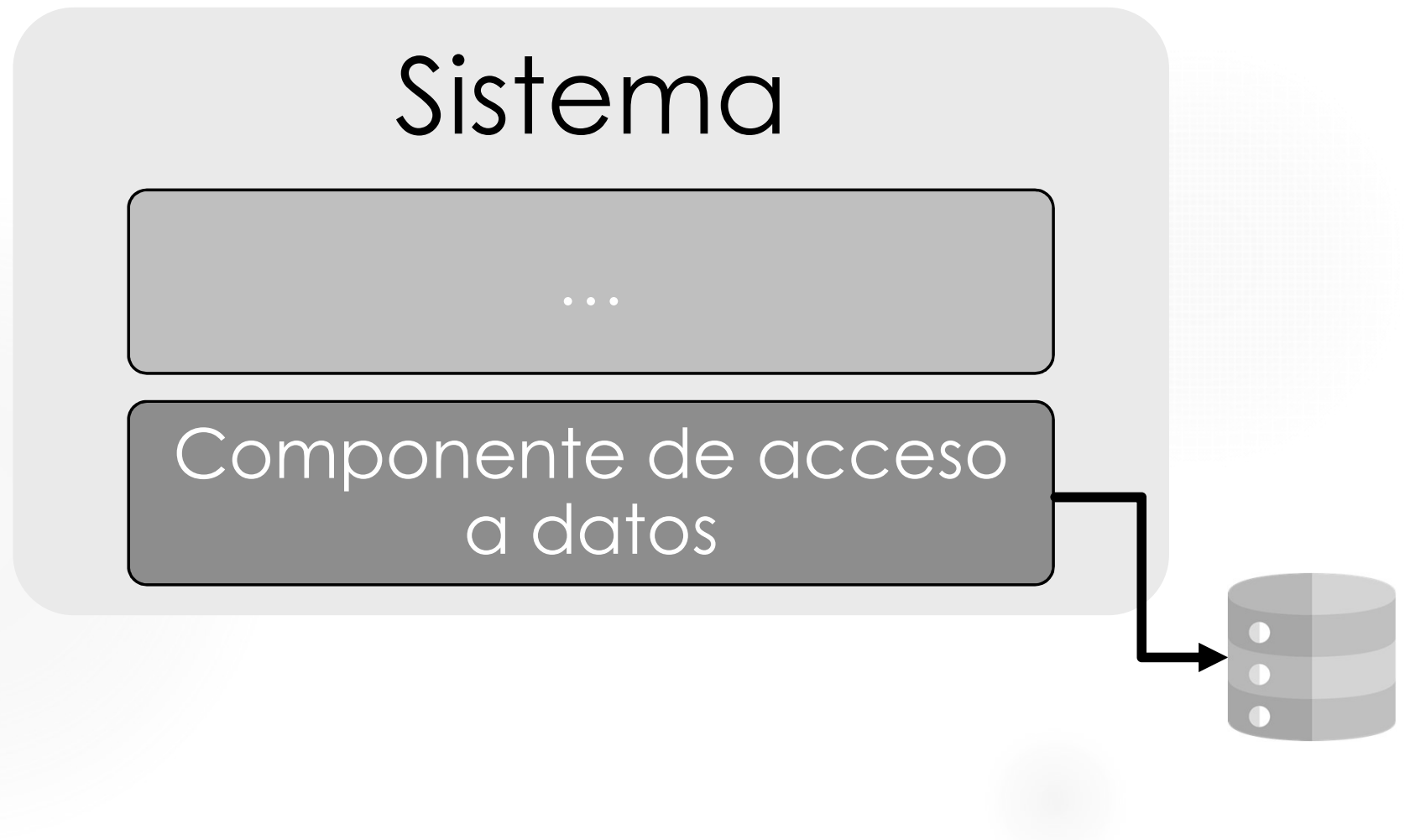
API'S WITH ASP.NET CORE



Object-Relational Mapping

ENTITY FRAMEWORK CORE

Arquitectura de sistema



Imaginemos el siguiente escenario

SqlServer

- `SELECT TOP 10 * FROM usuarios;`

MySQL

- `SELECT * FROM usuarios LIMIT 10;`

Oracle

- `SELECT * FROM usuarios WHERE rownum<=10;`

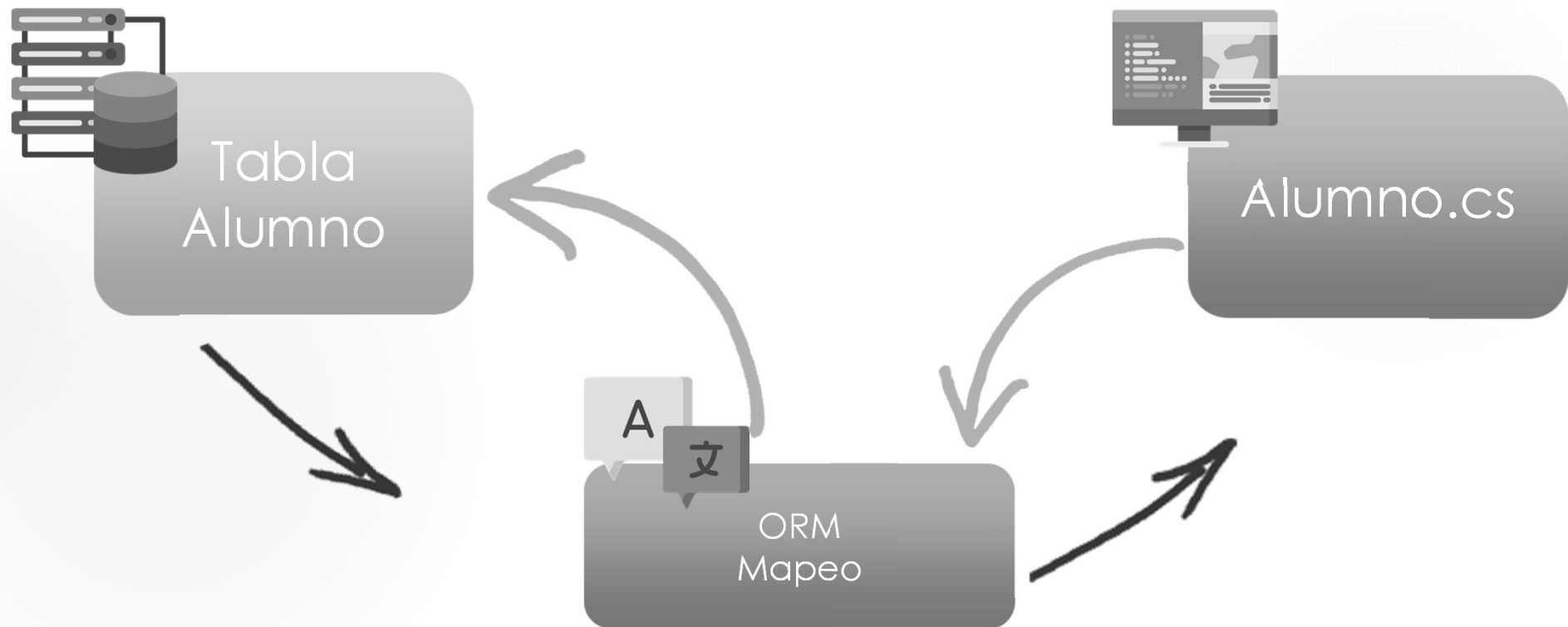
¿Qué es un ORM?

“

Es una técnica de programación que consiste en la transformación de las tablas de una base de datos, en una serie de entidades (Clases) que simplifiquen las tareas básicas de acceso a los datos para el programador.

”

¿Por qué es útil?



Ventajas de su uso



Independencia del motor de base de datos



Menos código, codificación más rápida y fácil

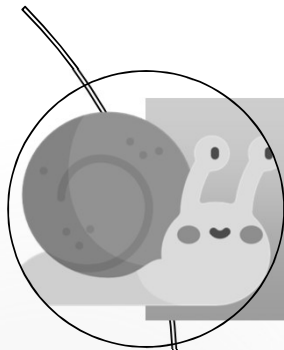


Aumento de seguridad en la capa de acceso a datos



Enfoque en la lógica de negocio

Algunas desventajas

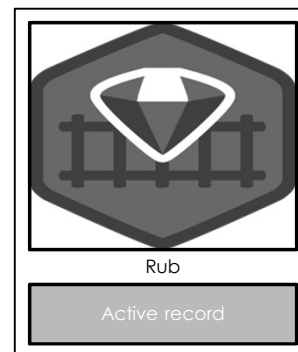
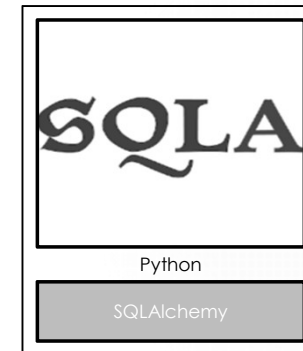
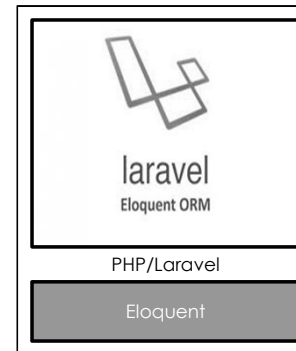
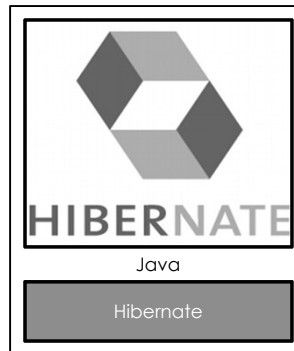
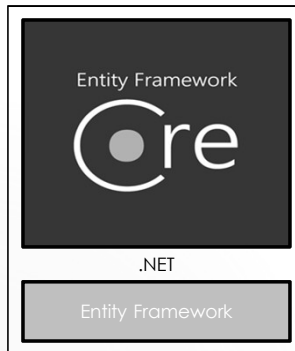


Puede ser complicado optimizar el código SQL generado

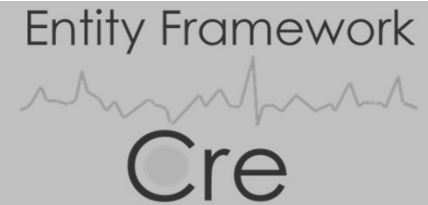


En algunos entornos puede ser demasiado complejo para el tamaño del proyecto

Ejemplos de ORM



Entity Framework Core



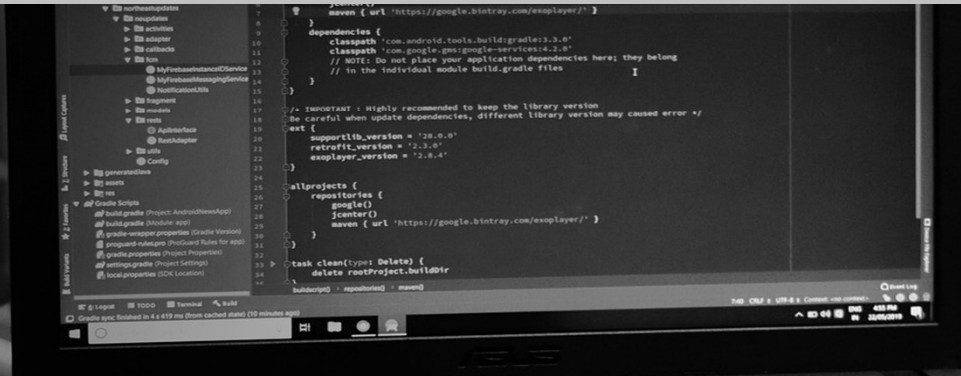
- Entity Framework Core es un asignador de base de datos de objeto moderno para .NET. Admite consultas LINQ, seguimiento de cambios, actualizaciones y migraciones de esquemas. EF Core funciona con una gran variedad de bases de datos, incluidas SQL Database (en el entorno local y Azure), SQLite, MySQL, PostgreSQL y Azure Cosmos DB.

<https://docs.microsoft.com/en-us/ef/>



Preguntas

Porque la práctica hace al maestro



Es hora de abrir Visual Studio y modificar nuestra aplicación