

# TeoriaTema6DisenodelaPersistenci...



yavae



Ingeniería del Software



3º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingeniería Informática  
Universidad Politécnica de Valencia



Si tu ordenador tiene más años que los  
que dura tu carrera: toca cambio.



**Katana GF66**

¿Necesitas un portátil para estudiar pero  
eres de los que se echa una partidita para  
desconectar?



# Encuentra **el trabajo** de tus sueños

Participa en retos y competiciones de programación



Escanéame y  
obtén más info!!

## TeoríaT6 Diseño de la Persistencia

Yavae

### 1. ¿Qué es el patrón de acceso DAO?

El patrón de acceso DAO es un patrón de diseño de software que se utiliza para separar la lógica de acceso a datos de la lógica de negocio de una aplicación.

### 2. ¿Cuáles son los componentes principales del patrón DAO?

Los componentes principales del patrón DAO son:

- DataAccessObject (DAO): Con el que se consigue un acceso transparente a la fuente de datos.
- DataTransferObject (DTO): representa el objeto portador de los datos. El DAO puede recibir y devolver datos mediante este objeto.
- BusinessObject (BO): objeto de la capa de negocio que necesita acceder a la fuente de datos, para leer o almacenar.
- DataSource: implementación de la fuente de datos

### 3. ¿Cuál es el objetivo principal del patrón DAO?

El objetivo principal del patrón DAO es aislar la lógica de acceso a datos de la lógica de negocio, lo que permite cambiar fácilmente la forma en que se almacenan y se recuperan los datos sin afectar a la lógica de negocio de la aplicación.

### 4. ¿Cuáles son las ventajas de utilizar el patrón DAO?

Entre las ventajas de utilizar el patrón DAO se pueden destacar las siguientes:

- Encapsulación. Los objetos de la capa de negocio no conocen detalles específicos de implementación del acceso a datos, ocultos en el DAO.
- Migración más fácil: migrar el sistema a un gestor de datos diferente supone cambiar la capa de DAO por otra.
- Menor complejidad en la capa de negocio al aislarse del acceso a datos.
- Centraliza el punto de acceso a datos.
- Separación de responsabilidades.
- Independencia de la base de datos.
- Facilidad para realizar pruebas y depurar el código, y para cambiar la forma de almacenar y recuperar los datos.

### 5. ¿Cuáles son las desventajas de utilizar el patrón DAO?

Entre las desventajas de utilizar el patrón DAO se podrían destacar las siguientes:

- Incluye una mayor complejidad.
- Hay que implementar mayor cantidad de código para ofrecer ese nivel de indirección adicional.
- Desde el punto de vista de la eficiencia se puede ralentizar el proceso.



## 6. ¿Qué es el patrón Repositorio + Unidad de Trabajo?

El patrón Repositorio + Unidad de Trabajo es un patrón de diseño de software que se utiliza para separar la lógica de acceso a datos de la lógica de negocio de una aplicación, se compone de dos partes: el repositorio y la unidad de trabajo.

## 7. ¿Cuáles son las funciones del repositorio y la unidad de trabajo en el patrón Repositorio + Unidad de Trabajo?

El repositorio es responsable de proporcionar una interfaz para interactuar con los datos almacenados en una base de datos, realizando operaciones CRUD. La unidad de trabajo es responsable de manejar las transacciones y asegurar que todas las operaciones se realicen correctamente.

## 8. ¿Qué es Entity Framework y cómo se relaciona con el patrón Repositorio + Unidad de Trabajo?

Entity Framework es una herramienta de acceso a datos que proporciona una interfaz para interactuar con una base de datos de manera sencilla. Con Entity Framework, se pueden utilizar los repositorios y unidades de trabajo para interactuar con los datos almacenados en una base de datos, logrando separar la lógica de acceso a datos de la lógica de negocio de la aplicación.

## 9. ¿Cuáles son las ventajas de utilizar el patrón Repositorio + Unidad de Trabajo?

Las ventajas de utilizar el patrón Repositorio + Unidad de Trabajo incluyen mejora en la separación de responsabilidades, facilidad para realizar cambios y mantenimiento de la aplicación, mejora en la escalabilidad y flexibilidad de la aplicación, y facilidad para realizar pruebas y depurar el código.

## 10. ¿Cuáles son las ventajas que nos aporta el uso de Entity Framework?

Entity Framework es un marco de trabajo de Microsoft que permite trabajar con bases de datos de manera más sencilla y eficiente. Algunas de las ventajas que nos aporta son:

- Abstracción de la base de datos: nos permite trabajar con entidades en lugar de escribir consultas SQL manualmente, lo que facilita el desarrollo y hace el código más legible.
- Manejo automático de la conexión: EF se encarga de abrir y cerrar la conexión con la base de datos, lo que nos ahorra tiempo y nos evita errores comunes.
- Caching automático: EF mantiene una copia de las entidades en memoria, lo que mejora el rendimiento al evitar tener que hacer continuamente consultas a la base de datos.
- Soporte para diferentes bases de datos: EF es compatible con varios sistemas de gestión de bases de datos, incluyendo SQL Server, MySQL, y SQLite.
- Generación automática de código: EF puede generar automáticamente el código necesario para trabajar con la base de datos a partir de un modelo, lo que nos permite centrarnos en el negocio.

### **11. Compara el patron repositorio con el patron DAO**

El patrón Repositorio y el patrón DAO son dos patrones de diseño de software que se utilizan para separar la lógica de acceso a datos de la lógica de negocio de una aplicación. Sin embargo, el patrón Repositorio ofrece una mayor abstracción, un mecanismo de unidad de trabajo y se centra en el manejo de objetos y en un nivel de abstracción de negocio, mientras que el patrón DAO se enfoca en una abstracción menor, no incluye unidad de trabajo y se centra en el manejo de tablas y filas de la base de datos y en un nivel de abstracción de acceso a datos.

### **12. Explica que es el DAL**

La interfaz IDAL (Interfaz de Acceso a Datos Abstracción) es un patrón de diseño que se utiliza para abstraer la lógica de acceso a datos de una aplicación. Esto nos permite crear una capa de abstracción entre la lógica de negocio y la base de datos, recibe el nombre de Capa de Acceso a Datos (Data Access Layer, abreviado DAL) DAL, su uso facilita la mantenibilidad y el escalado de la aplicación.

### **13. Explica las ventajas que aporta el uso conjunto del DAL con Entity Framework**

La combinación de la interfaz IDAL con Entity Framework nos permite utilizar las ventajas de ambos. La interfaz IDAL nos proporciona una capa de abstracción para trabajar con la base de datos, mientras que Entity Framework se encarga de implementar la lógica de acceso a datos. Esto nos permite trabajar con entidades en lugar de escribir consultas SQL manualmente, lo que facilita el desarrollo y hace el código más legible. Además, al utilizar la interfaz IDAL, podemos cambiar fácilmente la implementación de Entity Framework por otra tecnología si es necesario.