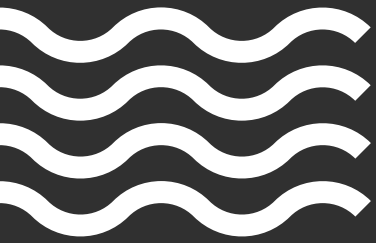


PATRÓN DE DISEÑO DAO

*Balan Fuentes Jair Alejandro
70003
Temas selectos de programación*



INDICE



¿QUÉ ES?



CARACTERÍSTICAS



USOS DEL PATRÓN DAO



VENTAJAS DEL DAO



CONCLUSIONES



1 ¿QUÉ ES?

Es un patrón que proporciona una abstracción sobre la capa de acceso a datos, encapsulando las operaciones de base de datos en una interfaz específica. De este modo, la lógica de negocio no interactúa directamente con la base de datos, sino a través del DAO.

Este patrón es particularmente útil en aplicaciones que pueden necesitar cambios en la fuente de datos sin modificar la lógica de negocio, permitiendo que los detalles de implementación del acceso a datos sean independientes de la lógica del sistema.

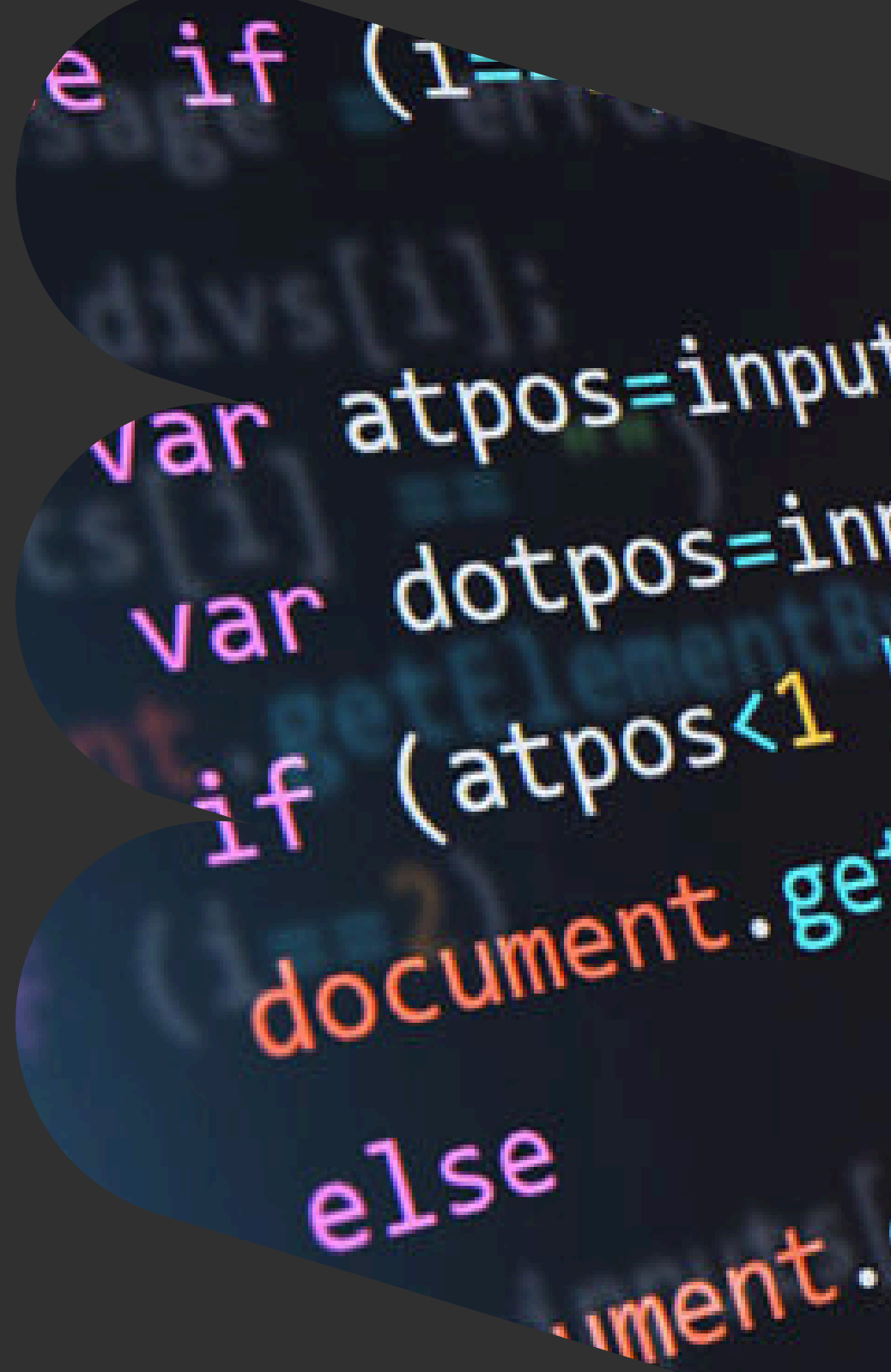


2

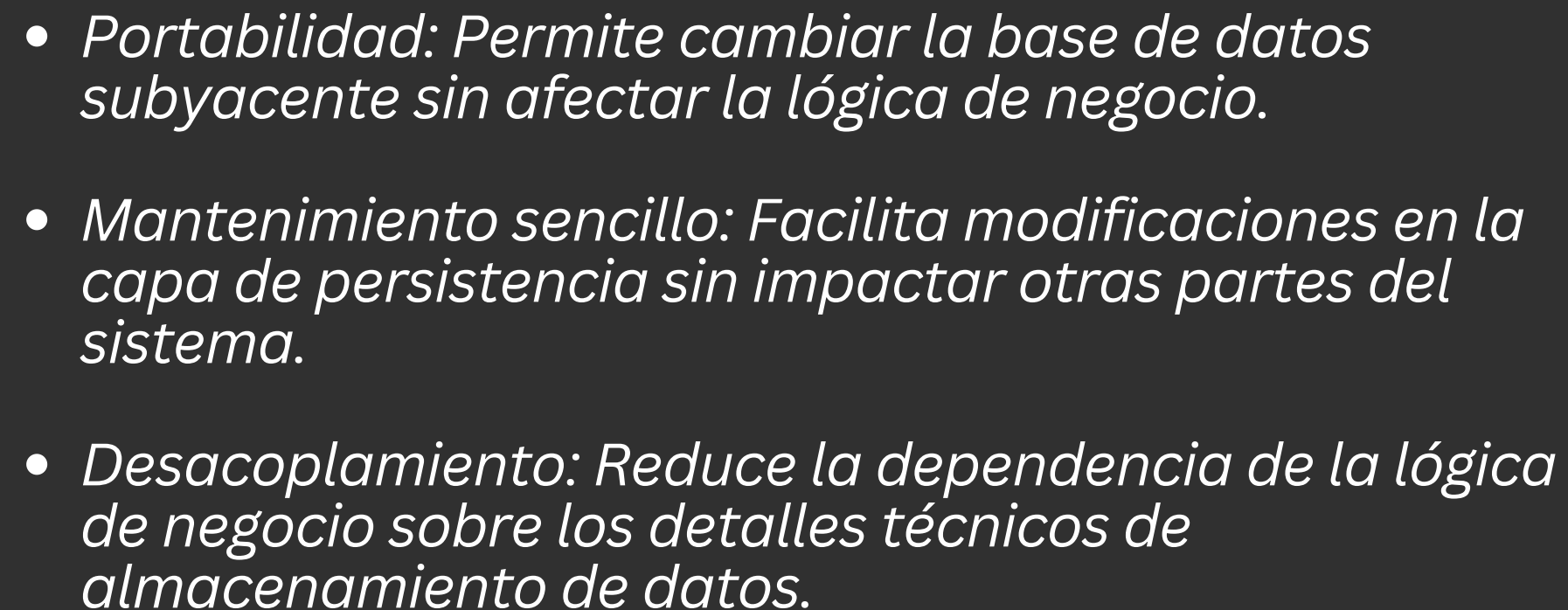
CARACTERÍSTICAS DEL DAO



- *Encapsulación: Oculta la implementación de acceso a datos, proporcionando una interfaz bien definida.*
- *Independencia: Separa la lógica de acceso a datos de la lógica de negocio, facilitando la evolución del software.*
- *Reutilización: Puede ser usado por diferentes partes de la aplicación sin redundancia, reduciendo la duplicación de código.*



CARACTERÍSTICAS DEL DAO



```


each: function(e, t) {
    var r, i = 0,
        o = e.length,
        a = M(e);
    if (n) {
        if (a) {
            for (; o > i; i++)
                if (r = t.apply(e[i], n))
                    return r;
        } else
            for (i in e)
                if (r = t.apply(e[i], n))
                    return r;
        } else if (a) {
            for (; o > i; i++)
                if (r = t.call(e[i], i, e[i]))
                    return r;
        } else
            for (i in e)
                if (r = t.call(e[i], i, e[i]))
                    return r;
    }
    return e;
},
is: b && !b.call("\uffff\u00a0") ? function(e) {
    return null == e ? "" : b.call(e);
} : function(e) {
    return null == e ? "" : (e + "").replace(/ /g, "");
},
isArray: function(e, t) {
    var n = t || [];
    return null != e && (M(Object(e)) && n.length > 0);
},
inArray: function(e, t, n) {
    var r;
    if (n < 0)
        n = t.length + n;
    for (r = 0; r < t.length; r++)
        if (t[r] === e)
            return r;
    return -1;
}

```



3

USOS DEL PATRÓN DAO

- Aplicaciones empresariales: Sistemas de gestión empresarial (ERP, CRM) que requieren un acceso eficiente y estructurado a bases de datos.
 - Sistemas con múltiples fuentes de datos: Aplicaciones que deben acceder a varias bases de datos diferentes o cambiar de proveedor de base de datos fácilmente.
 - Arquitecturas en capas: Es un componente clave en arquitecturas como MVC (Modelo-Vista-Controlador) y Arquitectura en Tres Capas, donde la separación entre datos, lógica y presentación es fundamental.
 - Aplicaciones escalables y mantenibles: En proyectos que requieren alta mantenibilidad y escalabilidad, DAO permite realizar cambios en la capa de persistencia sin afectar la lógica de negocio.
- 

4

VENTAJAS DEL DAO

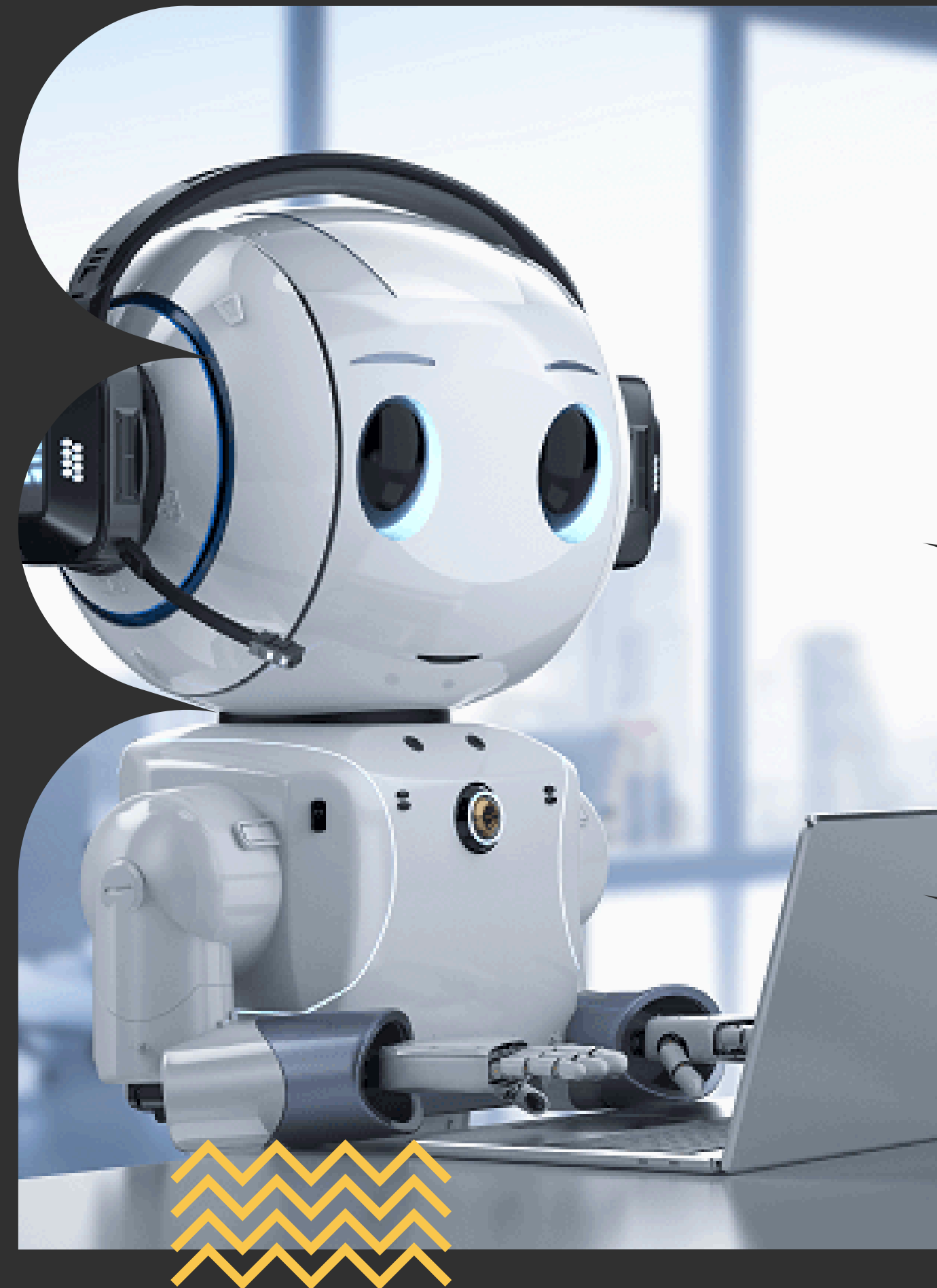
- Facilidad de migración: Permite cambiar de motor de base de datos sin afectar el código de la aplicación, utilizando interfaces para definir el comportamiento sin depender de una implementación específica.
- Eficiencia y rendimiento: Al estructurar adecuadamente el acceso a datos, se pueden optimizar consultas y transacciones para mejorar el rendimiento general del sistema.



4

VENTAJAS DEL DAO

- Separación de responsabilidades: Mantiene organizada la arquitectura del software, permitiendo que cada capa se enfoque en su función específica.
- Facilita pruebas unitarias: Se pueden simular accesos a datos sin una base de datos real, usando objetos de prueba como mock objects.
- Mayor seguridad: La lógica de negocio no interactúa directamente con la base de datos, reduciendo el riesgo de SQL Injection y otros ataques.



¿POR QUÉ USARLO?

- Modularidad: Facilita cambios en la lógica de acceso a datos sin modificar otras capas.
- Escalabilidad: Permite crecer la aplicación sin generar dependencias rígidas, asegurando que la lógica de negocio no dependa de la tecnología de base de datos específica.
- Reducción del acoplamiento: Evita que la lógica de negocio dependa directamente de la implementación de acceso a datos, permitiendo que se pueda reemplazar o actualizar sin afectar el sistema en su totalidad.
- Manejo de transacciones y conexiones: Puede gestionar eficientemente las conexiones a la base de datos, evitando fugas de memoria y mejorando el rendimiento.

DIFERENCIAS ENTRE DAO, LÓGICA DE NEGOCIO Y BASE DE DATOS

Componente	¿Qué hace?	¿Lo maneja DAO?
Lógica de Negocio	Aplica reglas, validaciones y procesos del sistema.	 No, eso lo maneja otro componente.
Lógica de Acceso a Datos	Conecta con la base de datos, ejecuta SQL, devuelve resultados.	 Sí, esto es lo que maneja el DAO.
Base de Datos	Almacena la información en MySQL, PostgreSQL, etc.	 Sí, el DAO interactúa con ella.

¡GRACIAS POR SU ATENCIÓN!

