

CRUD Estudiantes en Java con Patrones de Diseño

Universidad de las Fuerzas Armadas - ESPE
Carrera: Ingeniería de Software
Grupo 6
Fecha: 16/06/2025





Objetivo

Desarrollar una aplicación en Java para gestionar estudiantes usando:

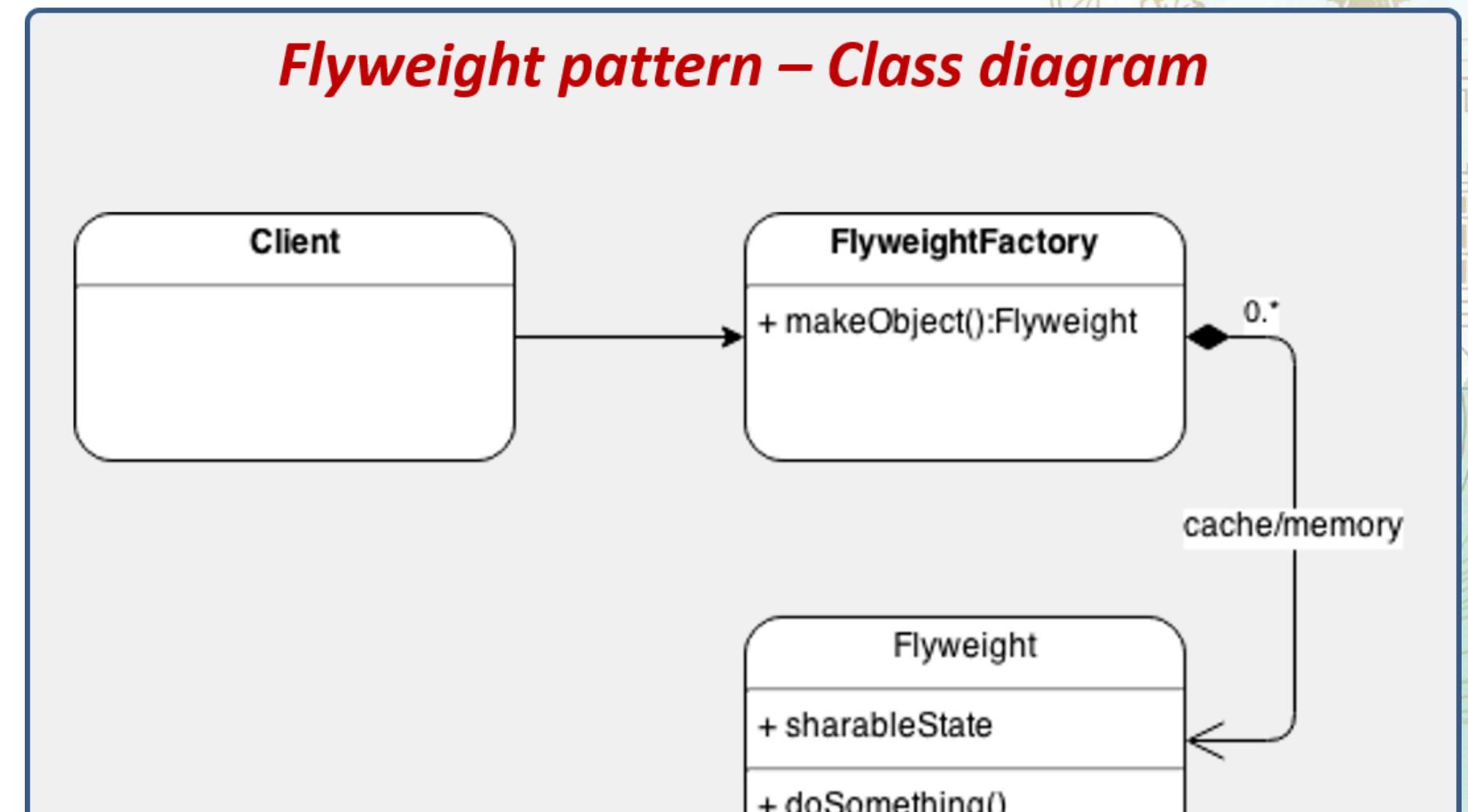
Patrones de diseño: Flyweight y Memento

Funcionalidades CRUD

- Agregar: Nuevo estudiante (ID, nombre, edad)
- Buscar: Por ID
- Actualizar: Editar datos
- Eliminar: Borrar estudiante
- Visualizar: Tabla dinámica con JTable

Patrón Flyweight

- Optimiza uso de memoria compartiendo nombres repetidos
- Implementación:
- `this.nombre = FlyweightFactory.getNombre(nombre);`





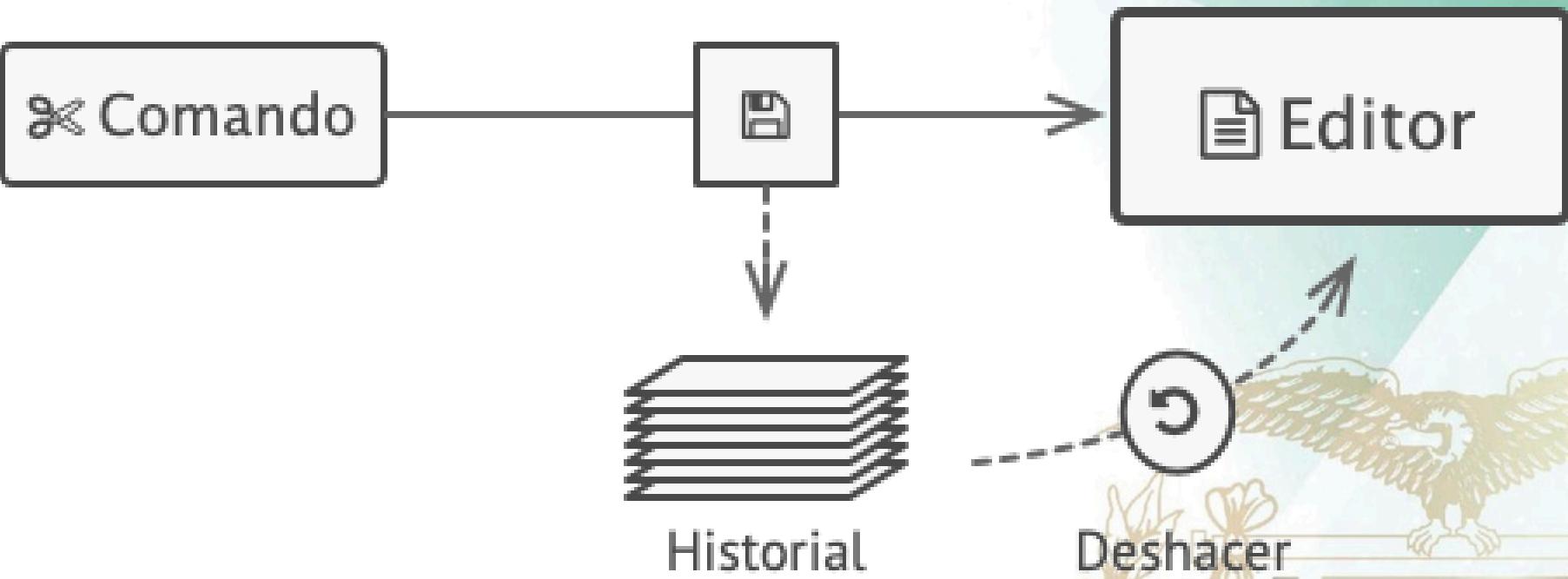
```
public Estudiante(int id, String nombre, int edad) {  
    this.id = id;  
    this.nombre = FlyweightFactory.getNombre(nombre); // Aplicación del  
patrón  
    this.edad = edad;  
}
```

Patrón Memento

- Permite deshacer cambios

Componentes:

- Memento: Guarda estado
- Originator: Crea/restaura
- Caretaker: Historial con pila (Stack)



```
package logica_negocio.memento;

import java.util.Stack;

public class Caretaker {

    private final Stack<Memento> historial = new Stack<>();

    public void guardar(Memento memento) {
        historial.push(memento);
    }

    public Memento deshacer() {
        return historial.isEmpty() ? null : historial.pop();
    }
}
```



Escalabilidad y Mejora

- Posible migración a interfaz web
- Integración de bases de datos
- Bajo acoplamiento
- Alta mantenibilidad



Estructura del Proyecto

- ProyectoEstudiantes/
- └── presentacion/
- └── logica_negocio/
- | └── memento/
- └── datos/
- | └── model/
- | └── repository/
- └── Main.java

Recomendaciones Futuras

- Persistencia con BDD o archivos
- Función Rehacer
- Validaciones robustas
- Pruebas unitarias (JUnit)
- Control de versiones con Git



Conclusión

- Sistema funcional con arquitectura clara
- Flyweight: Eficiencia en memoria
- Memento: Mejora experiencia del usuario



GitHub:

- https://github.com/KevinColoma/23305_G6_ADS

