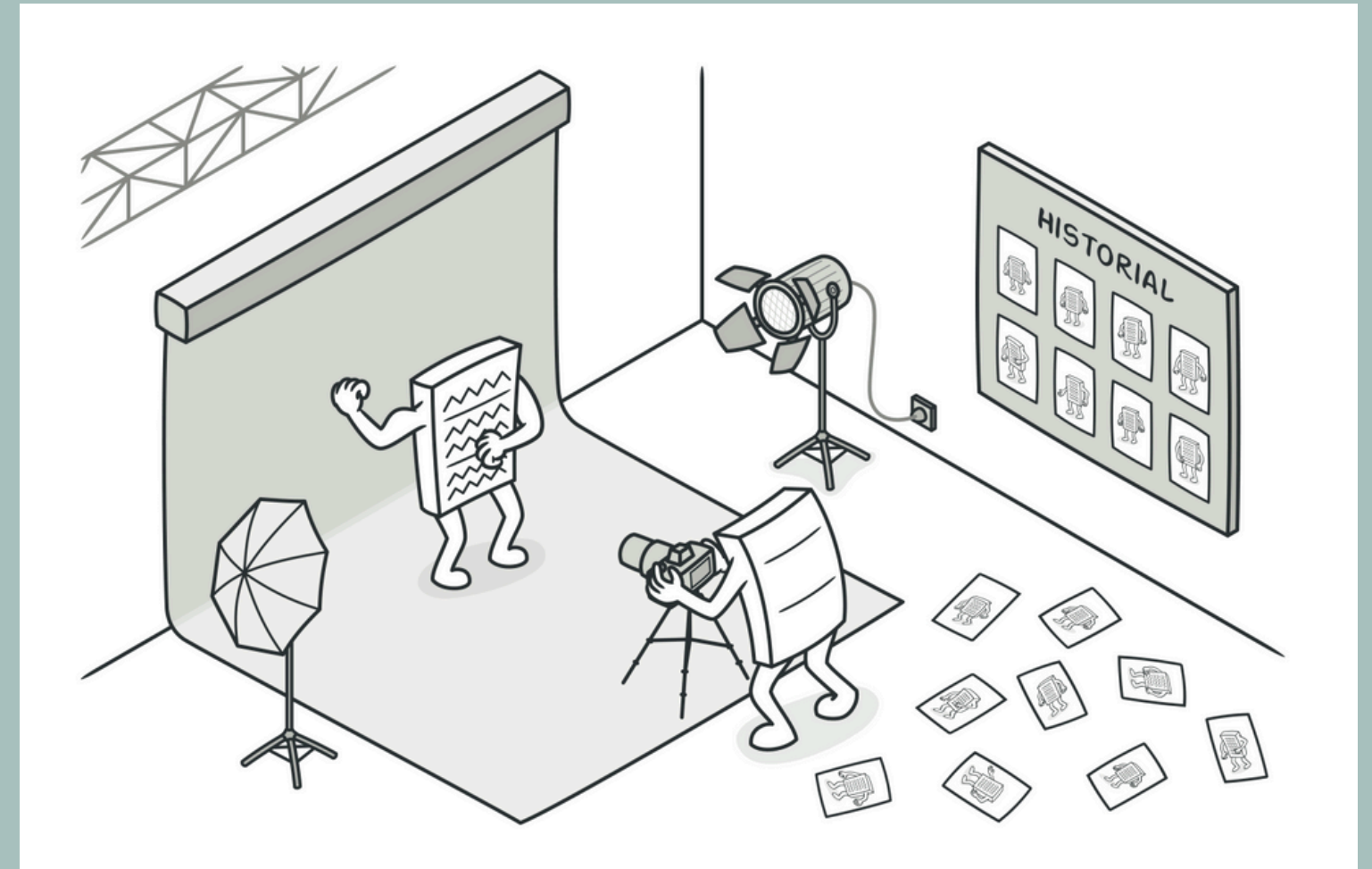


PATRON MEMENTO

Flavia Mendez

Resumen

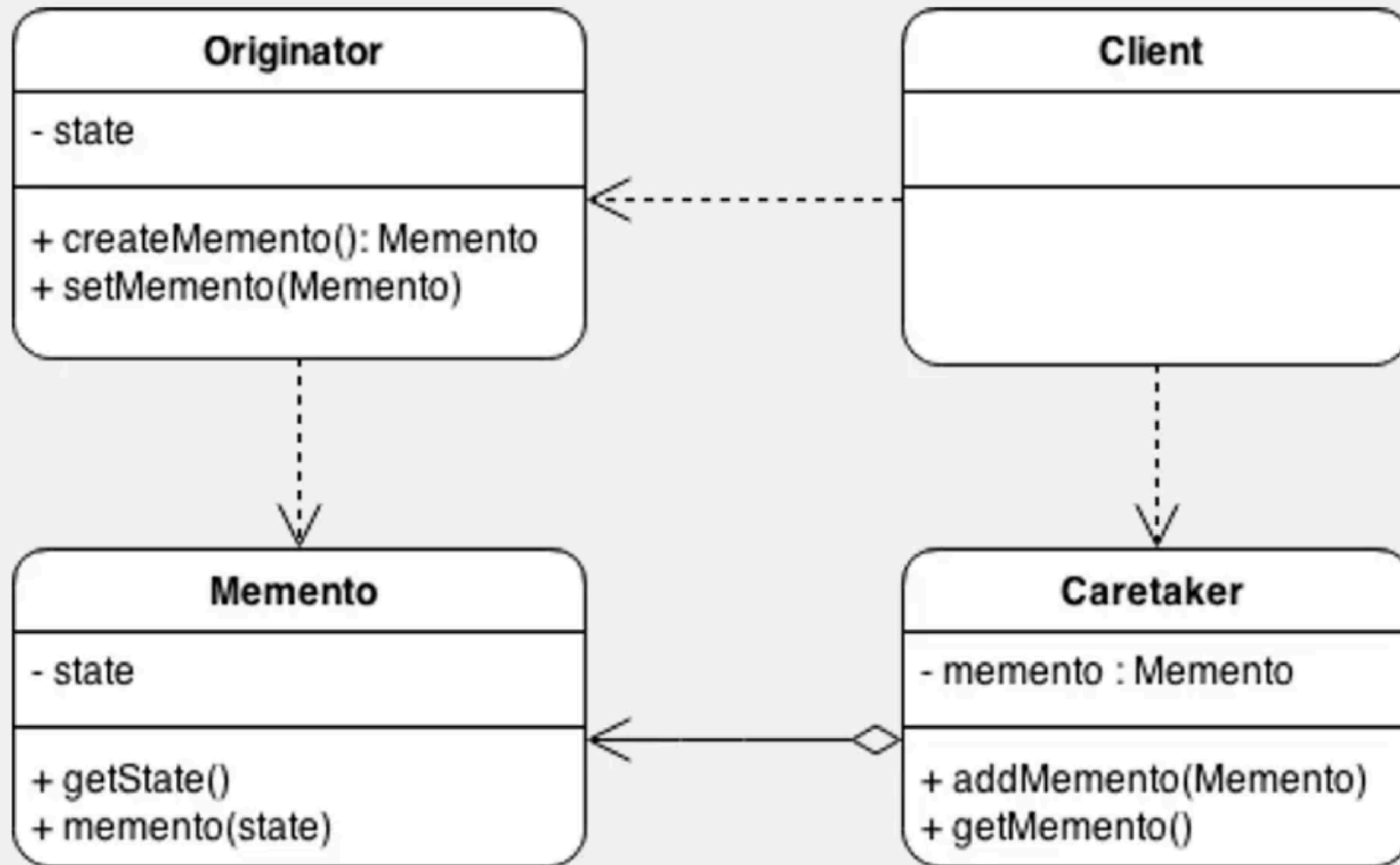
El patrón de diseño Memento es un patrón de comportamiento que permite capturar y restaurar el estado de un objeto sin violar su encapsulamiento. Es útil cuando se necesita deshacer cambios en un objeto y regresar a un estado anterior sin exponer los detalles internos de su implementación.



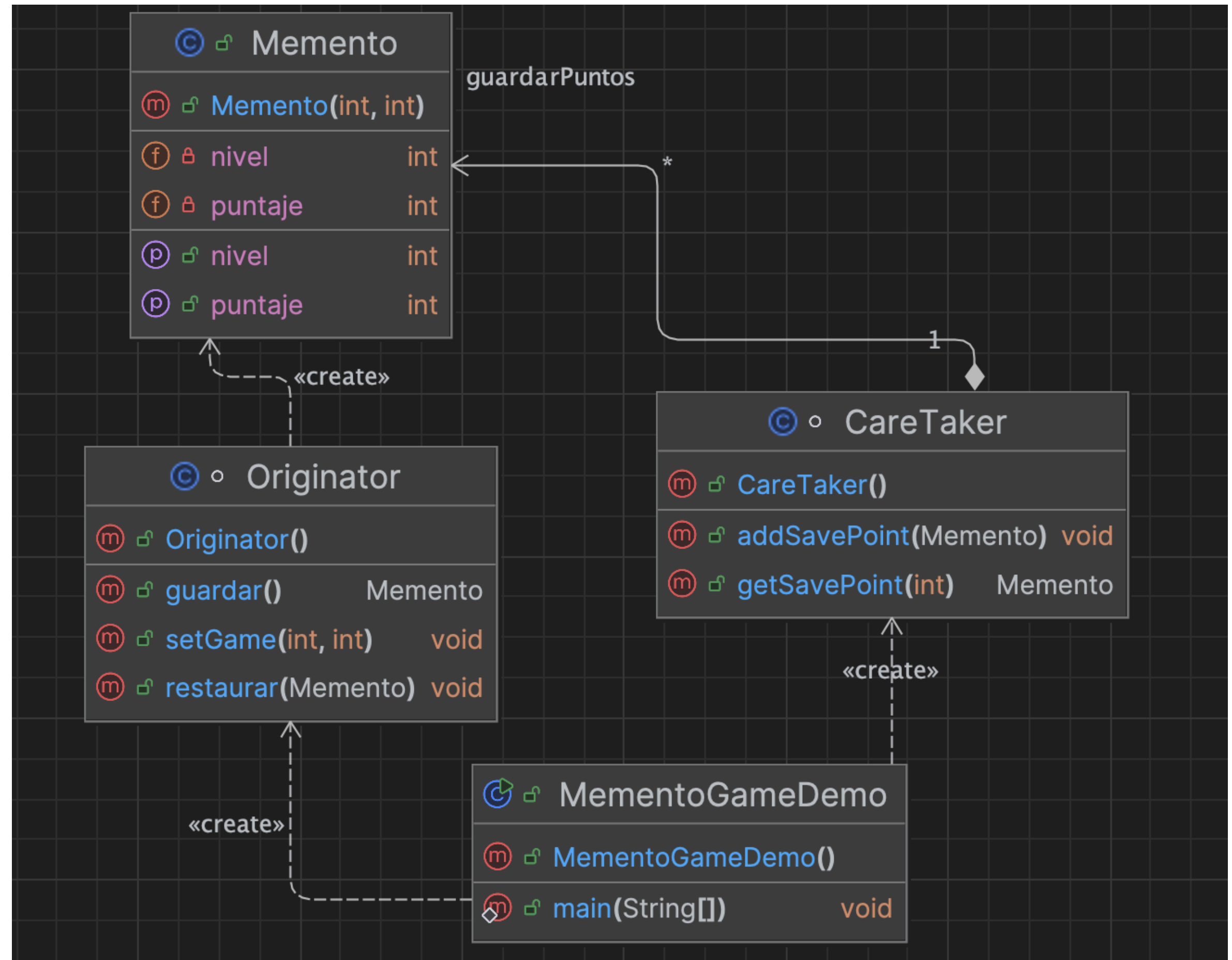
Estructura del patron

El patrón Memento consta de tres componentes principales:

1. **Originador:** Es el objeto cuyo estado queremos guardar y, en caso necesario, restaurar. Este componente tiene métodos para crear un memento (con su estado actual) y para restaurar su estado desde un memento.
2. **Memento:** Almacena el estado del Originador en un momento específico. Para proteger el encapsulamiento, el Memento no proporciona métodos para que el estado almacenado sea modificado o accedido desde fuera del patrón, salvo por el Originador.
3. **Caretaker:** Actúa como el guardián de los mementos. Su función es guardar y recuperar el memento del Originador sin conocer o manipular su contenido interno. El Caretaker solo administra los mementos sin interactuar directamente con los detalles del estado interno.



Ejemplo:



Código

```
//La clase memento almacena el estado del juego
public class Memento {

    private int nivel; //Nivel del jugador en el momento del guardado
    private int puntaje; //Puntaje del jugador en el momento del guardado

    //Constructor de Menemento que guarda el nivel y puntaje actual del jugador
    public Memento(int nivel, int puntaje) {
        this.nivel= nivel;
        this.puntaje= puntaje;
    }

    //Getter para obtener el nivel del guardado en el Memento
    public int getNivel() {
        return nivel;
    }

    //Getter para obtener el puntaje del guardado en el Memento
    public int getPuntaje() {
        return puntaje;
    }
}
```

```
// Clase Originator: Representa el estado actual del juego del jugador
class Originator {
    private int nivel; // Nivel actual del juego
    private int puntaje; // Puntaje actual del juego

    // Método para actualizar el estado actual del juego (nivel y puntaje)
    public void setGame(int nivel, int puntaje) {
        this.nivel = nivel;
        this.puntaje = puntaje;
        System.out.println("Nivel: " + nivel + ", Puntaje: " + puntaje);
    }

    // Método para crear un Memento que guarde el estado actual del juego
    public Memento guardar() {
        return new Memento(nivel, puntaje);
    }

    // Método para restaurar el estado del juego desde un Memento
    public void restaurar(Memento memento) {
        this.nivel = memento.getNivel();
        this.puntaje = memento.getPuntaje();
    }
}
```

```
import java.util.ArrayList;
import java.util.List;

// Clase CareTaker: Administra los Mementos para guardar y
cargar el progreso del juego
class CareTaker {
    private List<Memento> guardarPuntos = new ArrayList<>();
    // Lista de puntos de guardado (Mementos)

    // Agregar un nuevo punto de guardado a la lista
    public void addSavePoint(Memento memento) {
        guardarPuntos.add(memento);
    }

    // Obtener un punto de guardado específico de la lista
    public Memento getSavePoint(int index) {
        // Validación del índice para evitar errores
        if (index >= 0 && index < guardarPuntos.size()) {
            return guardarPuntos.get(index);
        } else {
            System.out.println("El punto de guardado no existe.");
            return null;
        }
    }
}
```



```
// Clase principal para demostrar el uso del patrón Memento en el juego
public class MementoGameDemo {
    public static void main(String[] args) {
        // Crear el juego (Originator) y el historial de guardado (CareTaker)
        Originator juego = new Originator();
        CareTaker historial = new CareTaker();

        // Simular el avance del juego y guardar estados
        System.out.println("Simulando el progreso del juego...");
        juego.setGame(1, 1000);
        historial.addSavePoint(juego.guardar()); // Guardar nivel 1, puntaje 1000
        System.out.println("Estado guardado: Nivel 1, Puntaje 1000");

        juego.setGame(2, 2000);
        historial.addSavePoint(juego.guardar()); // Guardar nivel 2, puntaje 2000
        System.out.println("Estado guardado: Nivel 2, Puntaje 2000");

        juego.setGame(3, 3000);
        historial.addSavePoint(juego.guardar()); // Guardar nivel 3, puntaje 3000
        System.out.println("Estado guardado: Nivel 3, Puntaje 3000");

        // Restaurar directamente el segundo guardado (nivel 2)
        System.out.println("\n== Intentando restaurar el Nivel 2 ==");
        Memento savedGame = historial.getSavePoint(1); // Recuperar el segundo guardado

        // Verificar y restaurar el guardado
        if (savedGame != null) {
            System.out.println("Punto de guardado encontrado. Restaurando el estado...");
            juego.restaurar(savedGame); // Restaurar al nivel y puntaje guardados
            System.out.println("Estado restaurado: Nivel " + savedGame.getNivel() + ", Puntaje " + savedGame.getPuntaje());
        } else {
            System.out.println("No se encontrado el nivel de guardado.");
        }
    }
}
```