Practica de laboratorio No. 7

Integrantes:

- Daniel Fernando Solarte Ortega
- Cristian David Quinayas Rivera

1.

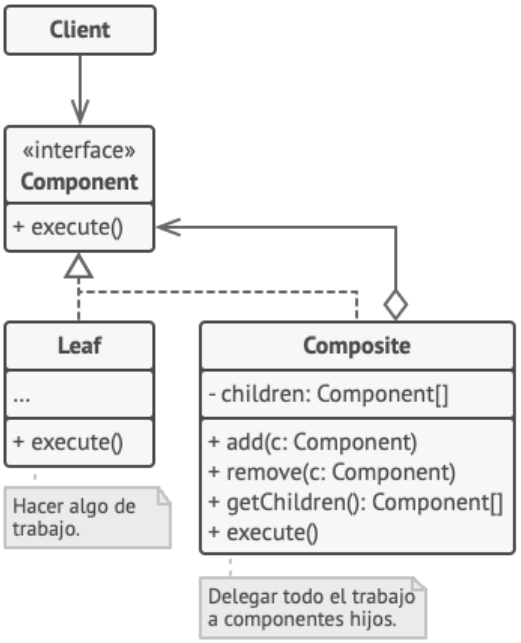
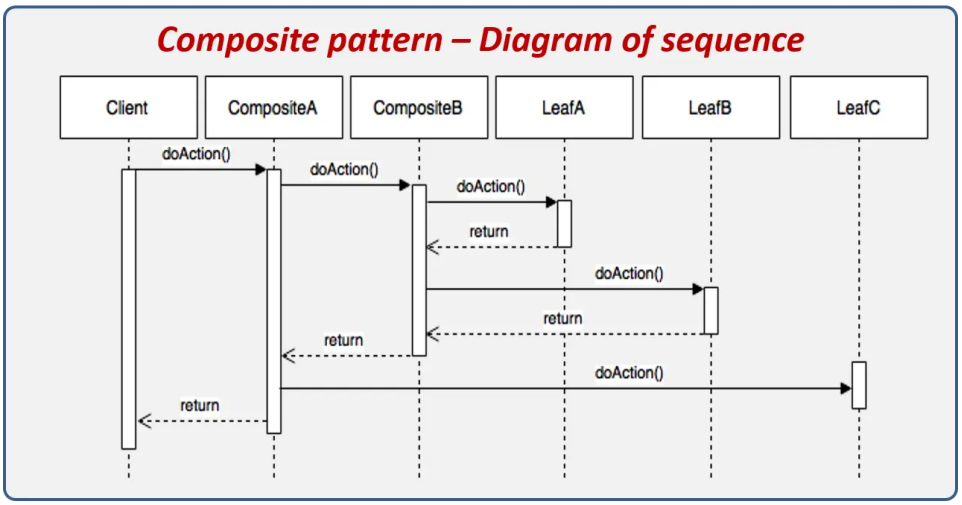
Patrón estructural: Composite	
Intención	Posibilitar la composición de objetos en estructuras de árbol (jerárquicas) para así poder utilizarlos como si fueran un objeto individual, esto con la ayuda de una interface, con el fin de que datos primitivos y compuestos dentro del árbol se traten como iguales.
Problema que soluciona	Cuando el modelo central de la aplicación que se esté desarrollando pueda ser representado en forma de árbol y sus hojas deben ser estructuras que pueden empaquetar otros tipos de estructuras que utilizan funciones en común.
Solución propuesta	Para que las partes que componen el árbol puedan utilizar métodos en común con el fin de facilitar funcionalidades se hace que tanto las hojas, como los contenedores de hojas implementen una misma interface de componentes la cual declara las funciones en común. También se utiliza una composición recursiva para que los contenedores puedan agregar más hojas en su interior.
Diagrama de clases	

Diagrama de secuencia	
Participantes	• Interface de componente. • Clase hoja (elemento más bajo en la jerarquía del árbol). • Clase contenedor (sirve como paquete, de forma que las ramas del árbol surgen de ellas).
Aplicabilidad	• Se utiliza cuando el modelo del problema a resolver sugiere que la mejor forma de hacerlo es con una estructura de objetos con forma de árbol. • Ayuda mucho cuando es necesario que tanto elementos simples como complejos sean tratados de la misma forma, o utilicen los mismos métodos.
Consecuencias	• Cuando existen clases cuya funcionalidad difiere mucho, es posible que crear una interface con la que estas puedan trabajar en común sea difícil, por lo que hacer que dicha interface sea demasiado general llevaría a problemas en el funcionamiento y la lógica de la solución. • El hecho de manejar un protocolo iterador y al mismo tiempo usar el patrón composite puede llegar a resultar ilógico, ya que el centro del patrón está en la capacidad de que el cliente pueda realizar operaciones en un objeto sin la necesidad de saber que hay muchos objetos dentro, cosa que dificulta el trabajo con iteradores. • El orden en que se agregar capas o decoradores concretos tiene relevancia en el resultado final del componente.