Memento

GestorHistorial

- mementosDeshacer: Deque<Memento>
- mementosRehacer: Deque<Memento>

+ deshacer(): Memento
+ guardar(Memento): void
+ rehacer(): Memento

Memento

- lienzoSnapshot: Lienzo

Composite

Lienzo

- capaActual: Capa
- capas: Capa []
- gestorComandos: GestorComandos
- gestorHistorial: GestorHistorial

+ agregarCapa(): void
+ cambiarOrdenCapa(int, int): void
+ cambiarVisibilidadCapa(int): void
+ capaSeleccionada(int): void
+ eliminarCapa(int): void

Capa

- figuras: Figura []
+ id: int
+ isSelected: boolean
+ isVisible: boolean
- trazos: Trazo []

+ agregarFigura(Figura): void
+ agregarTrazo(Trazo): void
+ dibujarTodo(): void

«abstract» Figura

- colorBorde: Color
- colorFondo: Color
- coordenadasFinales: Point
- coordenadasIniciales: Point

+ dibujar(Color, Point, Point, Color): void

GestorComandos

- pilaDeshacer: Deque<Command>
- pilaRehacer: Deque<Command>

+ deshacer(): void
+ ejecutarComando(Command): void
+ rehacer(): void

EstadoGestionCapas

+ moverMouse(int, int): void
+ presionarMouse(int, int): void
+ soltarMouse(int, int): void

Command

«interface» Command

+ deshacer(): void
+ ejecutar(): void

ComandoAgregarCapa

+ deshacer(): void
+ ejecutar(): void

ComandoEliminarCapa

+ deshacer(): void
+ ejecutar(): void

ComandoMoverCapa

+ deshacer(): void
+ ejecutar(): void

ComandoAgregarFigura

+ deshacer(): void
+ ejecutar(): void

ComandoEliminarFigura

+ deshacer(): void
+ ejecutar(): void

ComandoMoverFigura

+ deshacer(): void
+ ejecutar(): void

HerramientaDibujo

- estadoActual: EstadoHerramienta
- lienzo: Lienzo

+ cambiarEstado(EstadoHerramienta): void
+ moverMouse(int, int): void
+ presionarMouse(int, int): void
+ soltarMouse(int, int): void

Factory

Triangulo

+ dibujar(Color, Point, Point, Color): void

Rectangulo

+ dibujar(Color, Point, Point, Color): void

Circunferencia

+ dibujar(Color, Point, Point, Color): void

«abstract»
Figura

Color
Color
asFinales: Point
asIniciales: Point

or, Point, Point, Color): void

FabricaFigura

+ crearFigura(String): void

Trazo

- color: Color
- puntos: Point []

+ dibujar(Point [], Color): void

0..*

StatePattern

nCapas

nt, int): void
e(int, int): void
t, int): void

EstadoBorrado

+ moverMouse(int, int): void
+ presionarMouse(int, int): void
+ soltarMouse(int, int): void

EstadoDibujoLibre

+ moverMouse(int, int): void
+ presionarMouse(int, int): void
+ soltarMouse(int, int): void

EstadoDibujoFigura

+ moverMouse(int, int): void
+ presionarMouse(int, int): void
+ soltarMouse(int, int): void

ientaDibujo

EstadoHerramienta

(EstadoHerramienta): void
nt, int): void
e(int, int): void
t, int): void

1

«interface»
EstadoHerramienta

+ moverMouse(int, int): void
+ presionarMouse(int, int): void
+ soltarMouse(int, int): void

1