Ejercicio 2 c)

Para poder resolver el problema planteado, vamos a usar el patrón estado. Sin embargo, vamos a tener que realizar varias modificaciones en el diagrama para simplificar la implementación. Para empezar necesitaremos una interfaz que represente el estado actual de nuestro Ascensor el cual llamaremos *ElevatorState*. 3 clases, que implementarán dicha interfaz, se encargarán de representar el estado en el que se encuentra en un momento dado nuestro ascensor.

Para empezar, los atributos de abierto y en movimiento pueden quitarse, puesto que podemos saber, gracias a los estados, si está abierto, cerrado o en movimiento.

Ejercicio 3a)

Para resolver este problema utilizaremos el patrón Decorador que precisamente soluciona este tipo de problemas. Las ventajas respecto a crear tantas subclases es que, a pesar de que se tengan que crear también bastantes subclases en el patrón decorador, nos ofrece una manera más flexible de añadir nuevas características a un componente, además de no tener tanta cantidad de subclases como en la primera solución. Por último, el decorador nos ofrece cantidad de pequeños objetos transparentes.