

# Informe Ejercicio 1

## Principios SOLID:

Principio abierto-cerrado: Se muestra en las clases que cumplen el rol de Estado concreto, puesto que estas clases redefinen los métodos por defecto de la clase “EstadoHabitacion”.

Principio de inversión de la dependencia: Aparece en el uso de métodos de la clase abstracta en lugar de que cada clase tenga sus propios métodos. También aparece en la declaración de listas para almacenar habitaciones y resultados.

## Patrón empleado:

En este ejercicio hemos empleado el patrón estado debido a la posibilidad de modificar el comportamiento de un mismo objeto en diferentes momentos de este, es decir, según su estado.

Una habitación pasa por diferentes estados que se comportan de distinta forma. El comportamiento de cada estado distinto se recoge en una sola clase abstracta que, en nuestro caso, también recoge un comportamiento estándar. La clase “Hotel” recoge todos los métodos que se describen en el enunciado, pero simplemente llaman a los métodos de la clase “Habitación” y realizan un pequeño control de errores.

También empleamos el patrón fundamental de instancia única para definir los diferentes estados por los que pasa la habitación como “singletons”, así podemos compartir una sola instancia de cada estado entre todas las habitaciones creadas.

## Diagrama de clases:

\*Las imágenes se encuentran por separado también para facilitar su visualización.

En el siguiente diagrama UML se muestran las clases empleadas en el ejercicio y sus respectivos roles en el patrón de diseño.



## Diagrama de estados:

\*Las imágenes se encuentran por separado también para facilitar su visualización.

En el siguiente diagrama dinámico se muestran los estados por los que pasa una habitación y los métodos que hacen que una habitación cambie entre ellos.

