

UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

Facultad de Ciencias

Equipo NullPointerException:

Adrián Aguilera Moreno - 421005200

Diego Angel Rosas Franco - 318165330

Marco Antonio Rivera Silva - 318183583



Modelado y programación

Práctica 3

Menciona los principios de diseño esenciales de los patrones Decorator y Adapter:

Decorator

Principios de diseño esenciales:

- Hace que la composición de un objeto se pueda hacer de manera dinámica en tiempo de ejecución.
- Es un patrón abierto al cambio y cerrado a la modificación.
- Tiene un componente central(o componente concreto) sobre la cual se irán agregando decoradores (es decir, otros componentes).

Desventajas del patrón de diseño:

- Si quisieramos realizar modificaciones internas a las decoraciones más superficiales (*e.g.*, modificaciones al núcleo) tenemos que “desmontar” las decoraciones hechas posteriormente a la que queremos modificar, esto le aporta rigidez a nuestro código.
- El diseño se vuelve más complejo en términos de líneas de código.

Adapter

Principios de diseño esenciales:

- Vuelve compatibles dos sistemas que antes no lo eran.
- El usuario no percibe que hay dos sistemas distintos funcionando como uno a través del Adapter.

Desventajas del patrón de diseño:

- Este patrón no optimiza lo que ya existe, solo parcha lo que hay y lo hace funcional (de funcionar), aunque esto no necesariamente implica que sea la mejor implementación.
- Mientras más se usa en el sistema, aporta viscosidad, lo cuál es algo negativo.

Instrucciones de instalación, compilación y ejecución.

Se dará por hecho que el usuario sabe moverse en terminal.

Requerimientos previos:

- Se debe contar con Java en su computadora. De preferencia la versión más reciente.

Ejecución del proyecto:

- Si está leyendo esto significa que desempaquetó con éxito el proyecto.
- Abra su terminal y diríjase a la ruta donde desempaquetó el proyecto.
- Una vez estando en la ruta `Practica03_NullPointerException`, diríjase a `Practica03_NullPointerException/src/fciencias/modelado/`
- Ejecute: `“javac WaySubXLasPizzasDeDonCangrejo.java”`, esto generará los `.class` del proyecto.
- Ejecute: `“java WaySubXLasPizzasDeDonCangrejo”`, esto ejecutará el proyecto mostrándole el menú solicitado para la practica.

Diagrama UML:

