



Práctica 03

Modelado y Programacion 2022-2

"Paw patrols"
Castellanos Palacios Gerardo 318245630
Vargas Gutiérrez Julieta 318341945

Marzo 2022

1. Teoría

1.1. Decorator

El patrón decorator adjunta responsabilidades adicionales flexibles a un objeto dinámicamente, es decir, utiliza la composición para ampliar la funcionalidad de un objeto.

- Desventajas

1. Es difícil eliminar una envoltura específica de la pila de envolturas además de que los decoradores pueden dar lugar a muchos objetos pequeños en nuestro diseño, y su uso excesivo puede ser complejo.

1.2. Adapter

Este patrón permite convertir la interfaz de una clase en otra interfaz que deseemos así mismo nos permite que objetos con interfaces incompatibles trabajen juntas.

- Desventajas

1. La complejidad del código va aumentado y puede resultar excesivo para bases de código pequeñas porque hay que introducir nuevas interfaces y clases.

2. Práctica

Descripción de la práctica

Debido a los temas vistos de *Decorator* y *Adapter* se implementaron dos restaurantes, uno de de baguettes que tendrá un menú personalizale donde se podrá escoger el tipo de pan y los ingredientes extras y al final mostrará el el ticket donde se verá la descripción del pedido (en esta parte usamos decorator). El siguiente restaurante es una cadena de pizzas (aquí usamos adapter) donde se adapta el ticket de compra para que sea estándar.

Incovenientes

No hubo mayores inconvenientes en está práctica, solo tuvimos que analizar bien el problema para ver como los patrones trabajarían juntos.

Compilación y Ejecución

Descomprimir la carpeta zip, colocarse en la carpeta src y en la consola ejecutar

```
javac Main.java
```

En la siguiente linea

```
java Main.java
```