

01/05/18

Facultad de Ciencias

Ciencias de la Computación

Modelado y Programación

Proyecto 1

Integrantes del equipo:

1. David Bautista Cázares
2. Edgar Gante Araujo
3. Carlos Alexis Álvarez de la Rosa

Justificación de los patrones utilizados durante el proyecto:

Observer

El patrón *Observer* define una dependencia entre objetos para que cada vez que un objeto cambie su estado, se notifique a todos sus dependientes. Este patrón fue utilizado en el proyecto para implementar toda la parte de contabilidad de las tres torterías e informarle al jefe su inventario de ingredientes general.

Decorator

El patrón *Decorator* le permite cambiar dinámicamente el comportamiento de un objeto en tiempo de ejecución envolviéndolo en un objeto de una clase de decorador. Este patrón fue utilizado para implementar los ingredientes y las tortas simples e integrales.

Adapter

El patrón de adaptador le permite envolver un objeto incompatible en un adaptador para hacerlo compatible con otra clase. Este patrón fue utilizado para poder adaptar la tortería a la sandwichería.

Iterator

En la programación orientada a objetos, el patrón de *Iterator* es un patrón de diseño en el que se utiliza un iterador para atravesar un contenedor y acceder a los elementos del contenedor. Presenta una forma de acceder a los elementos de un objeto sin exponer la presentación subyacente. Nos fue muy útil para implementar el menú.

Builder

Builder permite crear diferentes versiones de un objeto mientras que evita la contaminación del constructor. Es muy útil cuando puede haber más de una variación de un objeto. O cuando hay muchos pasos involucrados en la creación de un objeto. Este patrón nos sirvió para crear las tortas predefinidas utilizando los ingredientes del decorator, ya que no era necesaria la personalización porque solo la piden y listo.