Introduccion

Los patrones de diseño describen la solución a problemas que se repiten una y otra vez en nuestros sistemas, de forma que se puede usar esa solución siempre que haga falta. Capturan el conocimiento que tienen los expertos a la hora de diseñar. Ayudan a generar software "maleable" (software que soporta y facilita el cambio, la reutilización y la mejora).

Que es un patron de diseño

Un patrón de diseño es una solución repetible a problemas típicos y recurrentes en el diseño del software. Son soluciones basadas en la experiencia y que se ha demostrado que funcionan. No son un diseño terminado que puede traducirse directamente a código, sino más bien una descripción sobre cómo resolver el problema, la cual puede ser utilizada en diversas situaciones.

Eric Gamma define los patrones de diseño como:

"Un patrón de diseño es una descripción de clases y objetos comunicándose entre sí para resolver un problema de diseño general en un contexto particular."

Clasificacion

Hay un listado de los principales patrones de diseño, entre ellos se encuentran los 23 patrones descritos en el libro de "Design Patterns – Elements of Reusable Object-Oriented Software", los cuales nombraremos a continuación:

✓ Patrones Creacionales

Los patrones creacionales ayudan a resolver los problemas relacionados con la creación de instancias de una clase, y así separar la implementación del cliente de la de los objetos que se utilizan. Nos ayudan a abstraer y encapsular su creación.



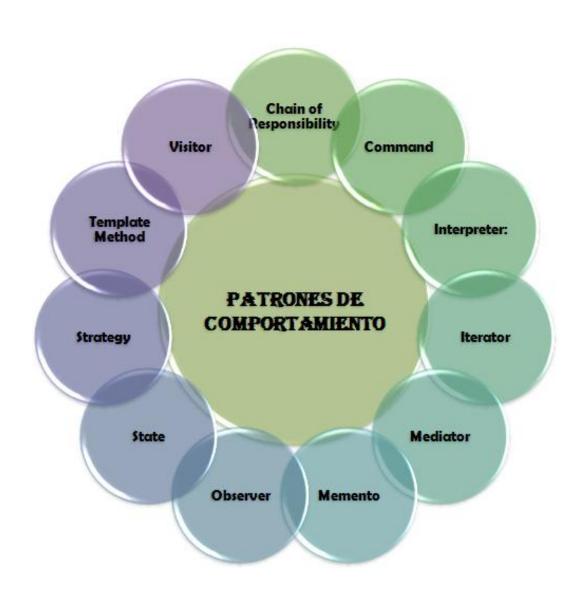
✓ Patrones Estructurales

Utilizados para crear clases u objetos que incluidos dentro de estructuras más complejas.



✓ Patrones de Comportamiento

Se utilizan a la hora de definir como las clases y objetos interaccionan entre ellos.



En nuestro proyecto aplicamos varios patrones de los cuales fueron:

FRONT CONTROLLER

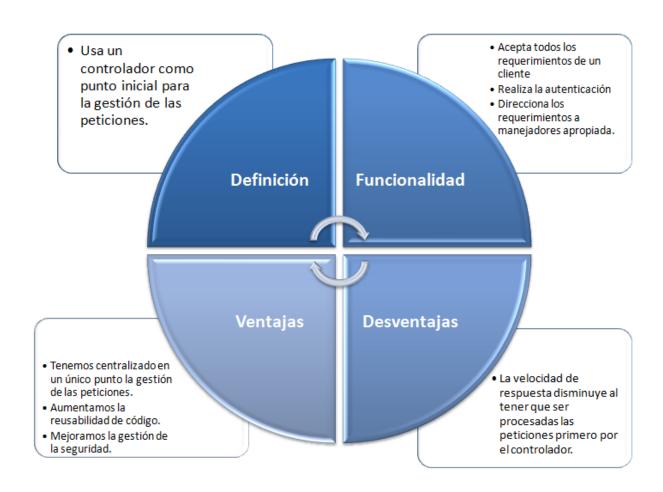
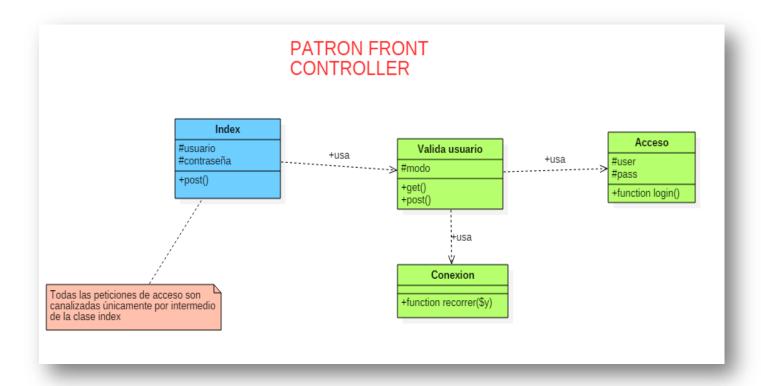


Diagrama UML



OBSERVER

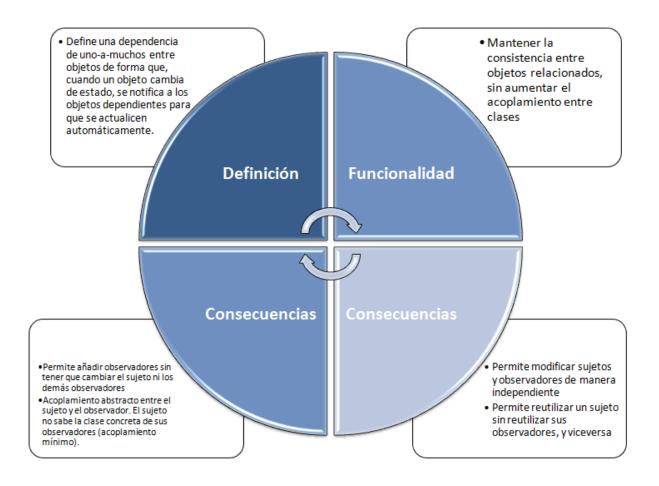
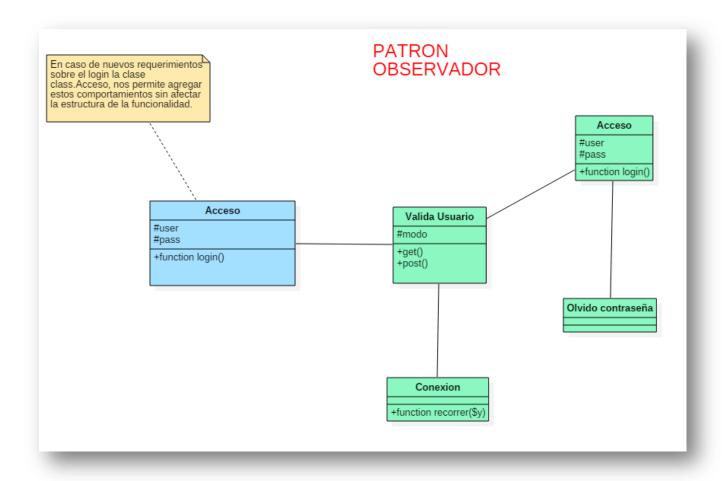


Diagrama UML



Singleton

Asegurar que una clase sólo tiene una instancia y proporcionar un punto global de acceso a ella.

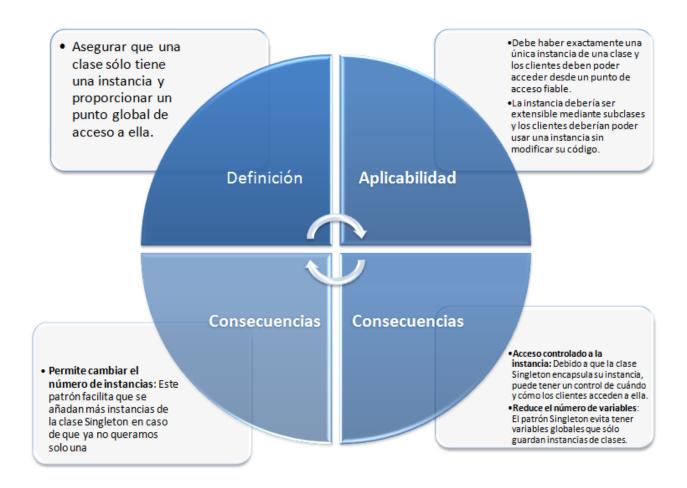


Diagrama UML

