

PHP Laravel

Clase 2

Ejemplos patrones de diseño más utilizados

PATRÓN DE DISEÑO

Patrones de creación

- Factory
- Prototype
- Singleton

Singleton

Garantiza que una clase sólo tenga una instancia, y proporciona un punto de acceso global a ella.

Factory

Consiste en utilizar una clase constructora dedicada a la construcción de objetos de un subtipo de un tipo determinado

Prototype

Especifica los tipos de objetos a crear por medio de una instancia prototípica, y crea nuevos objetos copiando de este prototipo.

PATRÓN DE DISEÑO

Patrones estructurales

- Adapter
- Decorator
- Facade

Adapter

Convierte la interfaz de una clase en otra distinta que es la que esperan los clientes. Permite que cooperen clases que de otra manera no podrían por tener interfaces incompatibles.

Decorator

Añade dinámicamente nuevas responsabilidades a un objeto, proporcionando una alternativa flexible a la herencia para extender la funcionalidad.

Facade

Proporciona una interfaz unificada para un conjunto de interfaces de un subsistema. Define una interfaz de alto nivel que hace que el subsistema sea más fácil de usar.

PATRÓN DE DISEÑO

Patrones de comportamiento

- Iterator
- Observer
- Strategy

Iterator

Proporciona un modo de acceder secuencialmente a los elementos de un objeto agregado sin exponer su representación interna.

Observer

Define una dependencia de una a muchos entre objetos, de forma que cuando un objeto cambie de estado se notifica y se actualizan automáticamente todos los objetos que dependen de él.

Strategy

En el patrón estrategia un comportamiento de una clase o un algoritmo de esta puede cambiar en tiempo de ejecución.

Creamos objetos que representan varias estrategias y un objeto contexto cuyo comportamiento varía dependiendo de los objetos estrategia. El objeto estrategia cambia el algoritmo que se ejecuta en el objeto contexto.