

# Patrones de diseño

Roni Guillermo Apaza Aceituno

Universidad Nacional de San Agustín

*rapazaac@unsa.edu.pe*

October 19, 2018

# Introducción

- Qué es un patrón?
- Qué es diseño?
- Es útil?

## patrón, na

Del lat. *patrōnus*; la forma f., del lat. *patrōna*.

En acep. 7, u. t. el m. para referirse a una mujer.

1. m. y f. Defensor, protector.
2. m. y f. Santo titular de una iglesia.
3. m. y f. Santo elegido como protector de un pueblo o congregación religiosa, profesional o civil.
4. m. y f. Dueño de la casa donde alguien se aloja u hospeda.
5. m. y f. [señor](#) (|| persona a la que sirve un criado).
6. m. y f. [patrono](#) (|| persona que emplea trabajadores).
7. m. y f. Persona que manda un pequeño buque mercante o una embarcación de recreo.
8. m. Modelo que sirve de muestra para sacar otra cosa igual.
9. m. Metal que se toma como tipo para la evaluación de la moneda en un sistema monetario.
10. m. Planta en que se hace un injerto.
11. f. Galera inmediatamente inferior en dignidad a la capitana de una escuadra.

- Se puede hacer software sin diseño?
- Rehacer otra vez la rueda.
- No aprender de errores.

- Diseñar software orientado a objetos es fácil?
- Diseñar software orientado a objetos y reusable es fácil?
- Por lo tanto hacer software no es fácil.

- Hacer un software capaz de evolucionar tiene que ser reusable.
- Por lo menos si queremos que tenga futuro.
- Será el caso de un software para las aulas universitarias?

- Lo único seguro de la vida es el cambio y la muerte.
- Debemos anticiparnos a los cambios en los requisitos.
- El diseño debe soportar esos cambios.

- Los patrones de diseño estan orientados a ese cambio.
- Al principio definimos que significa patrón.
- Un ejemplo para entenderlo mejor.



- Todos queremos ser expertos en un área.
- Todos queremos tener las herramientas para resolver cualquier situación.
- Todos queremos ser maestros.

- Primero aprendemos las reglas básicas.
- Ahora somos maestros?
- Desgraciadamente no.

- Necesitamos de ciertos principios.
- Ahora somos maestros?
- Desgraciadamente no.

- Necesitamos estudiar a otros maestros.
- Ahora somos maestros?
- Cuantos maestros existen?

- Se aplica las mismas reglas para el software.
- Logicamente por eso uno esta aqui.
- Si, al final hay que aprender varios patrones.

- Propuesto por la banda de los cuatro. ("Gang of four")
- Gamma, Helm, Johnson y Vlissides.
- Design Patterns. Elements of Reusable Object-Oriented Software.

- Un patrón es una solución a un problema con ciertas características.
- Como es una solución, lo hace recurrente para otras situaciones.
- Como elemento instructivo permite el entendimiento de las condiciones previas para aplicarlo a un problema actual.

- El patrón como elemento instructivo tiene un nombre.
- Los patrones son útiles para el reuso del diseño.
- También son útiles para las arquitecturas de software para conseguir éxito.



- Es esto un reflejo de la colaboración?
- Claro sin colaboración no hay avance.
- Sin comunicación no hay interacción.

- Usamos la experiencia de otros desarrolladores.
- Los otros desarrolladores hacen accesible estos conocimientos para los no expertos.
- Al interactuar con estos desarrolladores generamos una forma de comunicación.

- El lenguaje de patrones es el utilizado para comunicarnos.
- Este lenguaje se basa en un conjunto de nombres formando un vocabulario.
- Este lenguaje implícito en el diseño crea un mejor entendimiento.

- Acompañado de la documentación da una mejor comprensión.
- Como tal también pueden ser la base de un manual.
- Las prácticas exitos deben ser documentadas sistemáticamente.

- No solo basta con ser documentadas.
- También deben ser difundidas.
- Solo es documentación?

- Permite una reestructuración de todo el sistema.
- Esto independientemente si fue hecho con patrones o sin ellos.
- La reutilización es mucho mas de lo que aparenta.

- Los patrones de diseño soportan la reutilización de las arquitecturas de software.
- Esto permite la reutilización del código.
- Al mismo tiempo la reutilización del diseño.

- Todo cambia incluido el software.
- Hay que ser profeticos con respecto del software.
- Para esto sirve los patrones del diseño.



# Clasificación de patrones de diseño

- Patrones creacionales.
- Patrones estructurales.
- Patrones de comportamiento.

# Clasificación de patrones de diseño

- Patrones creacionales, concernientes a la creación de objetos.
- Patrones estructurales, tratan de la composición de clases y/o objetos.
- Patrones de comportamiento, muestran las formas de interacción entre las diferentes clases y objetos asignando responsabilidades.

# Patrones creacionales

- Factory Method.
- Abstract Factory.
- Prototype.

# Patrones creacionales

- Singleton.
- Builder.

# Patrones estructurales

- Adaptador
- Bridge
- Flyweight

# Patrones estructurales

- Glue
- Proxy

# Patrones estructurales

- Composite
- Wrapper

# Patrones de comportamiento

- Template Method
- Chain of Responsibility
- Command



# Patrones de comportamiento

- Iterator
- Mediator
- Memento

# Patrones de comportamiento

- Observer
- State
- Strategy

# Patrones de comportamiento

- Interpreter
- Walker

# Preguntas