



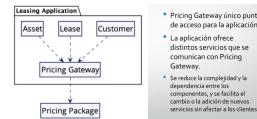
T7 : PATRONES I

GATEWAY

- Simplificar y encapsular lógica de acceso a sist. ext.
- Ej: interactuar con API

ventajas

- Abstracción complejidad
- Centralización lógica de acceso en un solo punto
- Facilita mantenimiento
- Gestión de errores

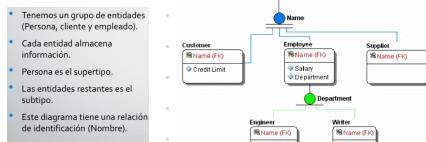


- ### desventajas
- Añade otra capa adicional que genera mayor sobre carga e consumo
 - Posible acoplamiento excesivo

PATRONES BÁSICOS

LAYER SUPER TYPE

- Org. y extracción de clases obj.
 - Para abordar problemas
 - Desventajas
- ventajas
- Duplicidad
 - Cohesión
 - Facilita comprensión Sist.
 - Promociona reutilización código
- Desventajas
- Zarzuela demasiado compleja (puede)
 - Requiere análisis cuidadoso de las relaciones entre clases



MAPPER

capa intermedia

- Separar lógica de Negocio \leftarrow de \rightarrow lógica acceso a datos
- transforma Obj dominio en Obj almacenam.

ventajas

- Abstracción
- Desacoplamiento entre Obj dominio y almacenam.
- Reutilización de obj (evita duplicados)

- ### desventajas
- Complejidad
 - Acoplamiento
 - Inconsistencia



- Objetos de dominio: Customer, Lease, Asset

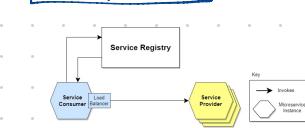
Registry

- Almacenamiento central

Ej: Web con varios servicios

conector

Registra y almacena obj necesarios



ventajas

- Centralización acceso
- Evita Ref. globales
- Facilita gestión config.

desventajas

- Cuellos de botella
- Complejidad
- Modularidad f al menor acoplamiento
- & Cohesión

Value Object

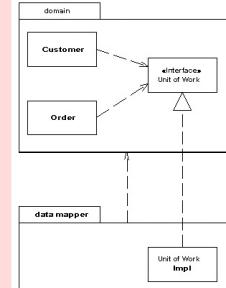
- Obj inmutable \Rightarrow valor se define por ATRs no por su entidad
- Obj es si tiene = campos, si no de si ? Atm es f

ventajas

- Cohesión
- Encapsulamiento
- Inmutabilidad

desventajas

- Puede generar duplicados e inconsistencias
- & complejidad
- & no clases
- Puede afectar al rendimiento



T7 : PATRONES I

PATRONES BÁSICOS

Money

Espec p Value Object

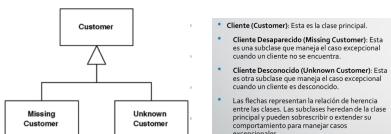
- problemas relacionados con dinero
(errores redondeo, precisión, = representación)
- ventajas
 - mejora precisión cálculos financieros
 - Evita errores redondeo
 - fácil conversión otros monedas
 - consistencia en toda la aplicación
- desventajas
 - requiere implementación adicional
 - Algunas operaciones pueden ser complejas



SPECIAL CASE

- Manejar situaciones excepcionales de forma organizada elegante

- | | |
|---|---|
| ventajas <ul style="list-style-type: none"> manejo excepciones claridad código exitos condicionales complejas | desventajas <ul style="list-style-type: none"> complejidad código diseño cuidadoso y documentación |
|---|---|



Plugin

- extender / personalizar funcionalidad → sin mod. código base

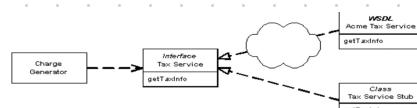
- | | |
|---|---|
| ventajas <ul style="list-style-type: none"> extensibilidad modularidad reutilización personalización | desventajas <ul style="list-style-type: none"> complejidad compatibilidad mantenimiento |
|---|---|



Service Stub

- Probar una parte del sistema q depende de un servicio externo q aún no está disponible
- ↓
simular comportamiento

- | | |
|--|---|
| ventajas <ul style="list-style-type: none"> facilita pruebas aisladas NO son pruebas de interac. real | desventajas <ul style="list-style-type: none"> Riesgo de sincronización |
|--|---|



Record Set

- representar y manipular conjuntos de datos tabulares (CSV) de manera estructurada y eficiente

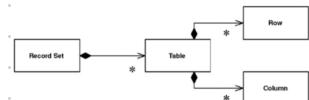
Ventajas

- Facilita manipulación de datos tabulares
- permite operaciones eficientes con conjuntos de registros
- proporciona abstracción →

desventajas

- puede introducir copia de abstracción
- Requiere una implementación y garantiza su uso efectivo

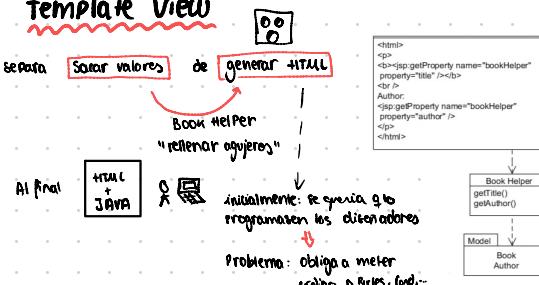
- Record Set:** Es la representación en memoria de los datos tabulares. Contiene uno o más filas.
- Table:** Representa una tabla de datos dentro del conjunto de registros. Contiene una o más filas.
- Row:** Representa una fila de datos dentro de la tabla. Contiene una o más columnas.
- Column:** Representa una columna de datos dentro de la fila.
- Este patrón es útil cuando necesitas realizar operaciones CRUD (Crear, Leer, Actualizar, Eliminar) en los datos almacenados en tu aplicación.



T7 : PATRONES I

Web Presentation Patterns

Template View

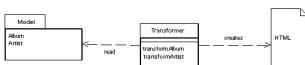


Transform View

Transformer → no partes de un HTML, sino q lo PRODUCES

Web que me da → HTML: QG gente
 XML: QG ordenado
 ↓
 Transformer

APP XML --> transformar --> HTML --> generar → Vistas Usuarias



TWO STEPS

- 1 ① det objetos + disparar vistas
- 2 PASOS ② combinar las

