

## Diseño de Sistemas Informáticos

4º Ingeniería Informática

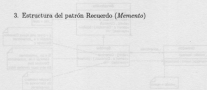
Examen Septiembre 2007

### Parte I: Cuestiones Generales

- Usando un diagrama de secuencia, presente la colaboración típica entre los participantes de un iterador (*Iterator*).

- Estructura del patrón Fábrica Abstracta (*Abstract Factory*)

- Estructura del patrón Recordo (*Memento*)



4. Usando un diagrama de secuencia, presente un ejemplo de interacción entre los participantes de un patrón Estado (State), en el que se produzca un cambio de estado.

Diagrama de Secuencia Interacción

4 Ejemplos Interacción

Examen Septiembre 2007

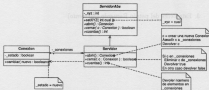
Parte II: Questiones Conceptuales

1. Usando un diagrama de secuencia, presente la colaboración típica entre los participantes de un patrón (Visitor).

5. Estructura del patrón Visitante (Visitor).

5 Estructura del patrón Visitante (Visitor Pattern)

6. Implemente en Java (únicamente) los métodos adicionales necesarios para que *ServidorA* soporte el patrón prototipo (Prototype) (utilice una clase adicional independiente)



## Parte II: Diseño

Se desea construir un Sistema de Gestión de Fotografías, que permita organizar y servir fotos para ser utilizadas en otras aplicaciones o directamente para ser visualizadas usando una interfaz de usuario. Una Foto proporciona un mapa de bits en formato JPEG y una serie de propiedades como su dimensión (alto y ancho), fecha de creación o una serie de palabras clave (opcionales) para describir la foto. Cada foto puede, además, incorporar opcionalmente un Pie (un pequeño título) o un Marco alrededor de la misma. Es interesante la posibilidad de poder definir un Mosaico a partir de varias fotos; en este caso, se debería poder configurar de manera flexible la Disposición de las fotos que componen un mosaico considerando, en principio, dos formas: en una *Rejilla* de  $n$  columnas y  $m$  filas o de una forma *Inteligente* intentando minimizar el espacio ocupado (aunque claramente surgirán nuevas maneras de disponer las fotos).

Aunque todas las fotos del sistema se ofrecen en formato JPEG, se desea poder utilizar fotos en otros formatos. En particular, se desea utilizar fotos en formato TIFF utilizando una biblioteca existente (MagickTIFF) que convierte mapas de bits entre los formatos TIFF y JPEG. Dado que la conversión es un proceso costoso computacionalmente sería deseable que en algunos escenarios se evite la conversión de la misma foto más de una vez.

El sistema incluye *Servidores* de Fotos que conocen colecciones de fotos concretas y ofrecen un servicio de búsqueda que devuelve una réplica de todas aquellas fotos que satisfacen un determinado *Criterio*: que contengan una determinada palabra clave, posteriores a una determinada fecha, etc. o combinaciones de estas. Es posible que un servidor de fotos mantenga una conexión con otro servidor auxiliar que pueda ser usado para complementar la respuesta en el caso de que el servidor no localice fotos que satisfagan el criterio del cliente. Los servidores pueden estar en distintos modos de funcionamiento: *Mantenimiento* (no pueden hacer búsquedas), *Online* (hacen búsquedas normalmente), *Saturado* (sólo devuelve la primera foto que cumple el criterio) u *Offline* (no hacen búsquedas). Solo se puede añadir/eliminar fotos a un servidor cuando se encuentra en modo mantenimiento. Inicialmente, un servidor se encuentra en modo offline hasta que explícitamente se cambia a modo online o mantenimiento. Estando en modo online, el servidor pasa a modo saturado si se atienden más de 10 peticiones en un segundo y no se vuelve a modo online hasta que el número de peticiones en un segundo baja de 2. Si en modo saturado la carga supera los 20 peticiones en un segundo se desconecta el servidor pasando a modo offline. Desde cualquier modo, se puede pasar explícitamente al servidor a modo offline.

Para organizar todos los servidores existentes, se define un *Metaservidor*, único en el sistema, que es responsable de conocer todos los servidores de fotos disponibles.

Se desea construir una interfaz de usuario lo más independiente de plataforma posible. Esta debe poder listar los servidores existentes a través del metaservidor, conectarse a un servidor de fotos concreto, debe permitir interrogar a dicho servidor con un criterio elegido dentro de una serie de criterios posibles y debe poder visualizar las fotos resultantes. En todo momento, la interfaz debe mostrar el número de fotos presentes en el servidor y este número debe actualizarse inmediatamente si el servidor incorpora o elimina fotos. Además, si el servidor se elimina del sistema (se retira del metaservidor) o se pone en modo offline, la interfaz debe desconectarse inmediatamente del servidor. Dado que la interacción con el sistema no se realizará únicamente desde esta interfaz de usuario, procure mantener la máxima independencia entre el sistema de gestión de fotografías y la interfaz de usuario.

Se pide:

- Diseño del sistema, justificando sus decisiones en base a los patrones empleados. Recuerde que el diseño no es únicamente un diagrama de clases (utilice todos los artefactos que considere necesario para describir adecuadamente la solución)