

Trabajo Práctico 1. Programación Orientada a Objetos

Puntuación

Puntaje Total: 100 puntos

Aprobación: 60 puntos

Fecha de entrega: 05/10/2015 - 23:55 Hs.

Condiciones de entrega

1. El presente trabajo práctico deberá resolverse en grupo de hasta tres (3) personas.
2. Entrega: Se realizará por medio del Campus Virtual de la UTN, en la tarea correspondiente al TP 1. La extensión del archivo será `.zip` o `.rar`, de acuerdo al programa de compresión usado. El nombre del archivo se consigue concatenando un prefijo del número del TP con los apellidos de los integrantes por orden alfabético y separados por guiones (Ej: para Pérez, Acevedo y González, el nombre será: TP1-Acevedo-Gonzalez-Perez.zip). Note que no hay espacios en blanco ni acentos en el nombre de archivo. Dentro del archivo de entrega, deben constar los siguientes:
 - Diagrama de clases conceptual, indicando la jerarquía de clases y relaciones entre las mismas. Puede utilizar Violet, UMLet, Dia, etc. Entregar en formato pdf. Nombre: `diagrama.pdf`
 - Fuentes Pharo Smalltalk: Se debe crear una categoría para todas las clases del TP, llamada TP1. Realizar un `#fileOut` de esa categoría y guardarla en un archivo `TP1.st`
 - Los casos de prueba se entregarán en un archivo de texto, no deben ser capturas de pantalla. Deberán cubrir diferentes resultados que puedan obtenerse según las funcionalidades solicitadas. Se enfatiza que se adjunten casos de prueba que sean claros, válidos y suficientes para poder probar el trabajo. Como sugerencia puede ir acumulando los casos en un Workspace (Playground) y luego grabarlo a disco con el nombre `casos-de-prueba.txt`.
 - Archivo de texto (`integrantes.txt`) con una línea para cada integrante en la cual figure el nombre del alumno/a y su dirección de email.
3. Penalización por entrega fuera de término: Si el trabajo práctico se entrega después de la fecha indicada, y hasta una semana tarde, tendrá una quita de 15 puntos. No serán recibidos trabajos luego de una semana de la fecha de entrega. Los trabajos que se deban rehacer/corregir fuera de la fecha de entrega tienen una quita de 30 puntos.

Descripción del problema: Sistema de Gestión de Revelado Digital de Fotografías

Un laboratorio fotográfico desea desarrollar un servicio Web de revelado digital de fotografías, el cual permitiría a sus clientes armar sus álbumes de fotos para ser impresos, incluyendo el envío a domicilio del álbum.

Entre otras cosas, el laboratorio necesita llevar un registro de los fotógrafos que realizan los trabajos para el mismo, y de este modo no perder la información del fotógrafo que tomó cada una de las fotos que se publican en el sitio. De los fotógrafos se necesita almacenar los datos personales (nombre y apellido, e-mail y dirección) y el nombre artístico. Cada fotógrafo es el encargado de subir sus fotografías para que estén disponibles en el sistema. Cada fotografía tiene una identificación, una descripción, y el nombre del archivo con la imagen asociada. Es importante que cada fotografía se asocie a un solo fotógrafo y éste sea el responsable de subirla, modificar su información, y eliminarla.

Un cliente que desea encargar un álbum, como primera medida, debe registrar sus datos personales. Si es un cliente registrado puede crear sus álbumes, donde cada álbum representa un pedido de impresión de fotografías. Para armar cada álbum, el cliente selecciona las fotos de las disponibles en el sitio (sacadas por los fotógrafos del laboratorio). Un álbum es creado inicialmente como *pendiente*, se lo identifica con un número y se lo describe con el nombre, fecha y hora de creación. Una vez conformado el álbum, se eligen el tamaño y el tipo de papel a utilizar para la impresión de las fotos del mismo. Los tipos y tamaños disponibles son los siguientes:

- Tipos de papel: brillo, mate.
- Tamaños: 10x15, 13x18, 15x20 (chico, mediano, grande).

Luego, al dar la orden de impresión, todas las fotografías de un álbum se imprimen con el tipo y tamaño de papel elegido por el cliente. Una vez impreso el álbum, además de la información que posee todo álbum, se almacena la fecha y hora de impresión, así como el costo total de impresión. El costo de impresión de una fotografía se define en base al tipo de papel y tamaño seleccionados. El costo total del álbum se determina en base al costo de cada fotografía multiplicado por la cantidad de fotos del álbum. Un álbum impreso no se puede modificar ni volver a imprimir.

Las funcionalidades de este sistema se pueden resumir como:

1. Incorporación y baja de fotógrafos con la información requerida.
2. Incorporación de clientes.
3. Carga, modificación y eliminación de fotografías al sitio.
4. Creación de un álbum. Esto implica seleccionar fotografías, tamaño y tipo de papel.
5. Impresión de álbum.
6. Consulta de cantidad de fotos sacadas por un fotógrafo.
7. Consulta de cantidad de álbumes impresos antes o después de una determinada fecha.
8. Consulta del costo de un álbum dado.
9. Consulta del tiempo promedio de demora en impresión de un álbum.

Implementación

Se solicita modelar el dominio del problema utilizando Orientación a Objetos, e implementar en **Pharo/Smalltalk** el sistema que dé respuesta a las funcionalidades solicitadas definiendo las clases y métodos siguientes, así como aquellas clases y/o métodos que considere necesarios.

Clase LaboratorioFotografico

Cuya instancia será responsable de gestionar a las instancias y clases restantes. Se solicita que implemente los siguientes métodos:

Método	Descripción
nuevoFotografo: unArreglodeDatos	Crea una instancia de fotógrafo e inicializa sus datos con la información que se recibe como argumento.
eliminarFotografo: nombreFotografo	Elimina el fotógrafo cuyo nombre y apellido están indicados por el argumento. Al eliminar el fotógrafo se deben eliminar todas las fotografías subidas por él que no estuvieran incluidas en algún álbum.
nuevaFoto: datosFoto de: nombreFotografo	Solicita al fotógrafo cuyo nombre se recibe como segundo argumento crear una instancia de foto con los datos que se reciben en el primer argumento. Devuelve la foto creada.
eliminarFoto: unaFoto de: nombreFotografo	Solicita al fotógrafo indicado que elimine la foto que se recibe como primer argumento (unaFoto).
buscarFotos: unTexto	Permite buscar fotos cuya descripción incluye (total o parcialmente) al texto pasado como argumento. Devuelve una lista con dichas fotos.
crearAlbum: listaFotos con: unTamaño enPapel: tipoPapel para: nombreCliente	Solicita al cliente de nombre nombreCliente que genere un álbum pendiente con las fotos en listaFotos, con un tamaño y tipo de papel especificados en los argumentos correspondientes. Devuelve el álbum creado.
impresionAlbum: unAlbum	Imprimir un álbum. Crea una instancia de álbum impreso a partir del álbum <i>pendiente</i> que se pasa como argumento.
cantidadFotosDe: nombreFotografo	Devuelve un entero que representa la cantidad de fotos que han sido subidas por el fotógrafo cuyo nombre se pasa como argumento.
albumesImpresosAntesDe: unaFecha	Devuelve una lista con las descripciones de los álbumes que han sido impresos antes de la fecha recibida como argumento.

albumesImpresosDespuesDe: unaFecha	Devuelve una lista con las descripciones de los álbumes que han sido impresos después de la fecha recibida como argumento.
albumesImpresosDesde: unaFecha hasta: otraFecha	Devuelve una lista con los álbumes que se han impreso en el período comprendido entre unaFecha y otraFecha.
costoDe: unAlbum	Devuelve el costo de impresión del álbum que se recibe como argumento.
demoraImpresion	Devuelve el tiempo promedio de la demora de impresión de los álbumes cargados en el sistema.
costoPromedioDesde: unaFecha	Devuelve el costo promedio de los álbumes impresos en el período comprendido entre unaFecha y la fecha actual.
buscarFotografoCon: unNombre	Devuelve el fotógrafo cuyo nombre y apellido coinciden con el argumento unNombre. Si no lo encuentra devuelve nil.
buscarClienteCon: unNombre	Devuelve el cliente cuyo nombre y apellido se reciben en el argumento unNombre.
verAlbum: unTexto de: unCliente	Retorna él o los álbumes cuyos nombres incluyen el texto indicado en el primer argumento y que pertenecen al cliente unCliente.
pendientesDesde: unaFecha	Retorna una lista con los álbumes que se encuentran pendientes desde unaFecha.
clientesPendientesDesde: unaFecha	Retorna una lista con los clientes que tienen al menos un álbum pendiente desde unaFecha.

Clase Fotógrafo

Método	Descripción
nuevaFoto: datosFoto	Crea una instancia de Foto e inicializa los datos con la información que se recibe como argumento. Devuelve la nueva foto.
eliminarFoto: unaFoto	Elimina el objeto foto pasado como argumento si dicha foto pertenece al fotógrafo (que recibe el mensaje).
actualizar: unaFoto con: datosFoto	Actualiza los datos de la foto indicada con el arreglo de datos pasado como segundo argumento.
verFotosDesde: unaFecha	Devuelve una colección con las fotos subidas por el fotógrafo que recibe el mensaje desde una determinada fecha.

Clase Cliente

Método	Descripción
crearAlbum: unNombre con: listaFotos de: unTamaño enPapel: tipoPapel	Devuelve un nuevo álbum pendiente cuyo nombre será unNombre, con las fotos incluidas en listaFotos, y asociándole un tamaño y tipo de papel especificados en los correspondientes argumentos. Devuelve el nuevo álbum creado.
buscarAlbumPor: unNombre	Devuelve el álbum cuyo nombre se recibe como argumento.
albumesPendientes	Devuelve una lista de los álbumes que el cliente que recibe el mensaje tiene en estado pendiente (un álbum que todavía no ha sido impreso).
costoPendientes	Devuelve el monto a pagar para imprimir todos los álbumes que tiene pendientes.
masViejo	Devuelve el álbum pendiente con fecha más antigua. Si no tiene pendientes retorna nil.

Clase Álbum

Método	Descripción
costo	Devuelve el costo de imprimir todas las fotografías.
imprimir	Para los álbumes <i>pendientes</i> devuelve un nuevo objeto que representa un álbum <i>impreso</i> , y para los álbumes impresos produce un error.
crearCopiaPendiente	Devuelve un nuevo álbum pendiente, con el contenido del objeto receptor.
pendiente	Devuelve un valor booleano según el tipo de álbum.
impreso	Devuelve un valor booleano según el tipo de álbum.
agregar: unaFoto	Incorpora unaFoto al álbum. Si el álbum está impreso no se puede agregar.
cantidadFotos	Devuelve la cantidad total de fotos que contiene el álbum.
eliminar: unaFoto	Elimina la foto pasada como argumento. Si el álbum está impreso no se puede modificar.