

Programación 4

PARCIAL FINAL EDICIÓN 2008

Por favor siga las siguientes indicaciones:

- Escriba con lápiz
- Escriba las hojas de un solo lado
- Escriba su nombre y número de documento en todas las hojas que entregue
- Numere las hojas e indique el total de hojas en la primera de ellas
- Recuerde entregar su número de parcial junto al parcial

Problema 1 (30 puntos)

- a) Responda en no más de 3 líneas cada una de las siguientes preguntas
- i) ¿Cuál es el objetivo y qué actividades se realizan en la etapa de Análisis?
 - ii) ¿Qué es una agregación y qué tipos de agregaciones existen?
 - iii) ¿Qué es un evento del sistema?
- b) Se está desarrollando un sitio web con un sistema de suscripción a publicaciones electrónicas. Una descripción reducida del dominio y del caso de uso principal aparece a continuación.

El sitio tiene disponible un conjunto de publicaciones electrónicas. Estas publicaciones son revistas y artículos científicos que pueden o no formar parte de una revista. Por lo general una revista contiene entre 2 y 10 artículos, los cuales se ordenan según un criterio definido para cada revista. Toda publicación tiene un título que la identifica, una fecha de publicación y un costo de suscripción. De estar incluido en una revista, la fecha de publicación de un artículo es la misma de la revista en la que aparece (dado que el artículo no puede publicarse más de una vez). Suscribirse a un artículo es proporcionalmente más caro que suscribirse a una revista por lo que el costo de una revista es siempre menor o igual a la suma del costo de los artículos que contiene.

Existen usuarios que se registran en el sitio mediante un nombre (que lo identifica) y una contraseña. Estos usuarios se suscriben para acceder a las publicaciones electrónicas. Si un usuario se suscribe a una revista entonces no puede suscribirse a ninguno de los artículos de esa revista. Además, se les permite realizar consultas sobre los autores de los artículos publicados (un artículo tiene eventualmente muchos autores). Por cada autor que se consulta (por nombre o afiliación del autor) se lleva registro de la cantidad de consultas que realizó cada usuario. Esto sirve para realizar promociones luego con nuevos artículos de los autores más consultados.

Nombre	Suscripción a Publicaciones	Actores	Usuario
Descripción	El caso de uso comienza cuando el usuario desea registrarse en el sitio y suscribirse a publicaciones. Para ello, el usuario ingresa una dirección de mail y datos personales, así como elige un nombre de usuario y contraseña. En caso de existir un usuario con ese nombre, el sistema pide al usuario que seleccione otro, no debiendo recibir nuevamente los datos personales ni el mail. En caso contrario, el sistema registra el usuario y se procede a la suscripción. A continuación el usuario solicita información de las publicaciones a lo que el sistema devuelve el título, costo y descripción (generada en el momento) de cada publicación. El usuario selecciona una a una cada publicación a las que desea suscribirse y el sistema responde afirmativa o negativamente según pueda o no suscribirse a ella. Finalmente, si el usuario desea confirmar la suscripción,		

	el sistema le devuelve el costo total y envía un mail a la dirección del usuario. En caso que desee cancelar, el sistema anula la suscripción a las publicaciones seleccionadas pero deja registrado al usuario.
--	--

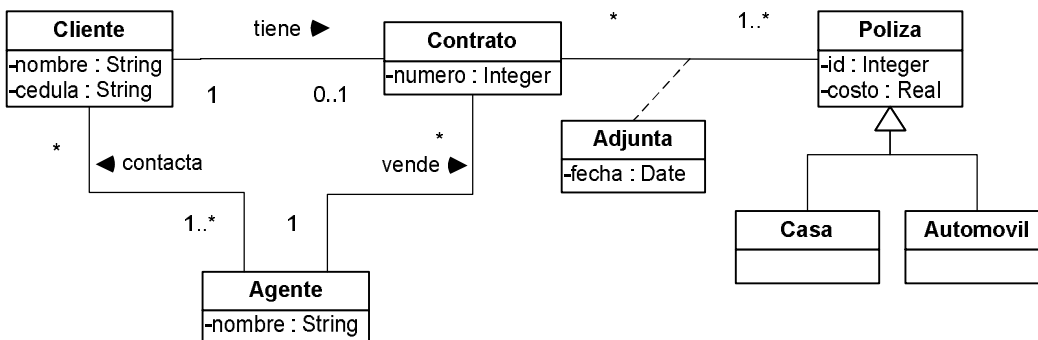
Se pide:

- i) Construir el Modelo de Dominio para la realidad descrita y presentarlo en un diagrama utilizando UML. Las restricciones deben ser expresadas en lenguaje natural y en OCL. Modelar exclusivamente en base a la información presente en la descripción y los casos de uso.
- ii) Realice un único Diagrama de Secuencia de Sistema para el caso de uso Suscripción a Publicaciones, incluyendo toda la información contenida en el mismo.

Problema 2 (40 puntos)

- a) Responda en no más de 3 líneas cada una de las siguientes preguntas
 - i) ¿A qué se le llama *visibilidad*?
 - ii) ¿Qué tipos de visibilidad se vieron en el curso?
 - iii) Describa dos tipos de visibilidad.

- b) Durante la construcción de un sistema de pólizas de seguros se generó el siguiente modelo de dominio. Básicamente, existen agentes que contactan a clientes para venderles pólizas de seguros. Los clientes firman un contrato con la empresa y en dicho contrato adjuntan las pólizas que deseen tener (casa, automóvil, etc.), las cuales pueden agregarse y quitarse del contrato a lo largo del tiempo.



Dentro de los casos de uso a soportar por el sistema, se encuentran las siguientes operaciones:

Nombre	Incluir póliza en contrato existente de un cliente
Operación	agregarPoliza(idP:Integer, ced:String)
Descripción	Agrega la póliza de identificador idP al contrato del cliente de cédula ced utilizando la fecha actual del sistema.

Nombre	Consultar clientes con póliza de automóvil
Operación	consultarPolizaDeAutomovil():Set(DataCliente)
Descripción	Retorna un conjunto de datavalues compuestos por el número de contrato y el nombre de cliente que posean alguna póliza de automóvil

Nombre	Dar de baja un contrato
Operación	bajaContrato(ced:String):Real
Descripción	Elimina el contrato del cliente de cédula ced y a su vez retorna el costo total de las pólizas que incluía dicho contrato.

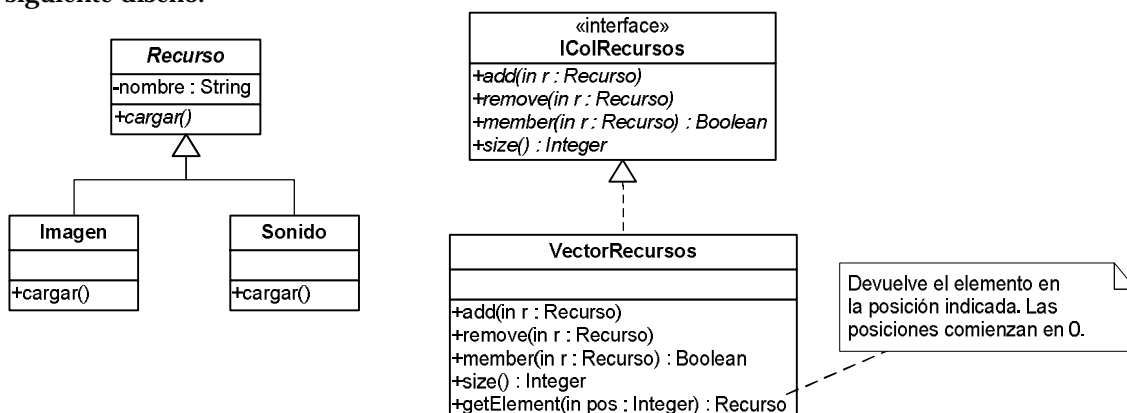
Se pide:

- i) Completar los contratos de las operaciones incluyendo las pre y post condiciones correspondientes. Considere exclusivamente la información dada en el problema y particularmente en la descripción de las operaciones.
- ii) Realizar un Diagrama de Comunicación para cada una de las operaciones respetando los contratos descritos. Explicitar la eliminación de instancias.
- iii) Realizar el Diagrama de Clases de Diseño incluyendo toda la información contenida en los diagramas.

Problema 3 (30 puntos)

PARTE A:

Se está construyendo la infraestructura para el futuro desarrollo de video-juegos. Usualmente los juegos deben manejar *recursos*, típicamente sonidos e imágenes. En este contexto se realizó el siguiente diseño:



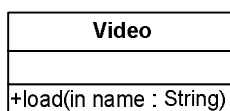
Se pide:

- i) Implementar en C++ la interfaz de un iterador de recursos para IColRecursos (siguiendo el patrón de diseño iterador). No es necesario incluir operaciones para remover elementos en el iterador.
- ii) Implementar en C++ una realización de la interfaz del iterador (definida en el punto i) para la clase VectorRecursos. No se puede modificar la clase VectorRecursos, con la salvedad de que se deberá agregar la operación que devuelve el iterador, ni asumir nada sobre su definición más allá de lo presentado en el diagrama.

Observación: No es necesario incluir directivas al preprocesador en el código.

PARTE B:

Luego se detectó la necesidad de manejar un nuevo tipo de recurso: el video. Dado que su manejo es complejo, se decidió utilizar una biblioteca implementada por terceros que provee la siguiente clase (no modificable):



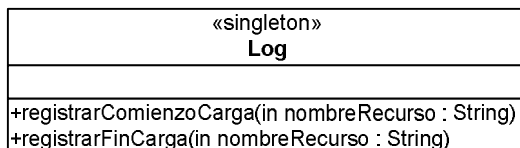
Se debe diseñar un mecanismo que permita incluir los videos dentro de la colección de recursos definida en la parte A.

Se pide:

- i) Realizar un DCD para el mecanismo mencionado aplicando patrones de diseño.
- ii) Explicar qué patrón(es) de diseño utilizó indicando (para cada patrón) las clases participantes y sus roles.

PARTE C:

Por último, se llegó a la conclusión de que era necesario registrar el tiempo que toma la carga de cada recurso, de forma de poder detectar posibles cuellos de botella. Para ello se cuenta con la siguiente clase auxiliar ya implementada:



Se debe evitar registrar los tiempos directamente desde los recursos concretos (Imagen, Sonido, etc.), dado que se espera el agregado de nuevos tipos de recursos que deberían sacar provecho de este mecanismo de forma transparente.

Se pide:

- i) Realizar el diseño completo (DCD y comunicaciones) del mecanismo mencionado aplicando patrones de diseño.
- ii) Explicar qué patrón(es) de diseño utilizó indicando (para cada patrón) las clases participantes y sus roles.