

Programación 4

PARCIAL FINAL EDICIÓN 2015

Por favor siga las siguientes indicaciones:

- Escriba con lápiz y de un solo lado de las hojas
- 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
- Está prohibido el uso de computadoras, tabletas o teléfonos durante el parcial

Problema 1 (35 puntos)

Parte a)

- Defina “estado de un objeto”.
- Explique y ejemplifique el concepto de polimorfismo.

Parte b)

FingFlix es una plataforma educativa que permite a los usuarios buscar y acceder a distintos contenidos que pueden ser individuales (texto, video y audio), ó colecciones (agrupación de los anteriores). De cada contenido se conoce su nombre (identificador), creador(es), fecha de creación y tema (matemática, informática, física, etc.). De los textos se conoce además su ISBN, de los videos su director y de los audios su formato. Las colecciones pueden agrupar contenidos individuales así como otras colecciones. Por ejemplo la colección “Curso de Matemática” puede contener otra colección “Audios de Geometría” y un texto “Cálculo I”. A su vez la colección “Audios de Geometría” tiene los audios “Clase1” y “Clase2”, y un texto “Índice de clases”.

De los usuarios se conoce su email (identificador), nombre y clave, con la cual pueden visualizar contenidos individuales. Cada vez que un usuario visualiza un contenido, se registra la fecha y posición en porcentaje (e.g. 30%) indicando dónde se abandona la visualización. Opcionalmente los usuarios podrán definir o actualizar una puntuación de [1..5] para cada contenido individual, lo que determinará la puntuación general (entre todos los usuarios) del contenido particular. La puntuación de colecciones se determina en base a la de sus contenidos.

Nombre	Visualizar Contenido
Actores	Usuario
Descripción	El caso de uso comienza listando los temas existentes. Luego que el usuario selecciona uno, se despliegan los nombres de todos los contenidos de ese tema. El usuario puede seleccionar cualquiera de ellos a modo de consultar los detalles y luego confirmar o no su visualización. Si no confirma se le pide seleccione otro nuevamente. Si se trata de un contenido individual se mostrarán los datos (nombre, fecha de creación, creador(es) y puntuación si tuviese). Si es una colección, se deberán listar los nombres de los contenidos de ésta, para luego realizar una nueva selección. El caso de uso finaliza cuando el usuario confirma la visuali-

	zacion de un contenido individual.
--	------------------------------------

Se pide:

- i. Considere la realidad y caso de uso planteado para realizar el Modelo de Dominio. Defina las restricciones correspondientes en lenguaje natural.
- ii. Realizar el Diagrama de Secuencia del Sistema correspondiente al Caso de Uso. Defina DataTypes en caso de ser necesario, y especifique el uso o no de la memoria del sistema.

Problema 2 (30 puntos)

Parte a)

- i. Definir el problema tipo del Patrón de diseño Observer, mostrar su estructura y los participantes.

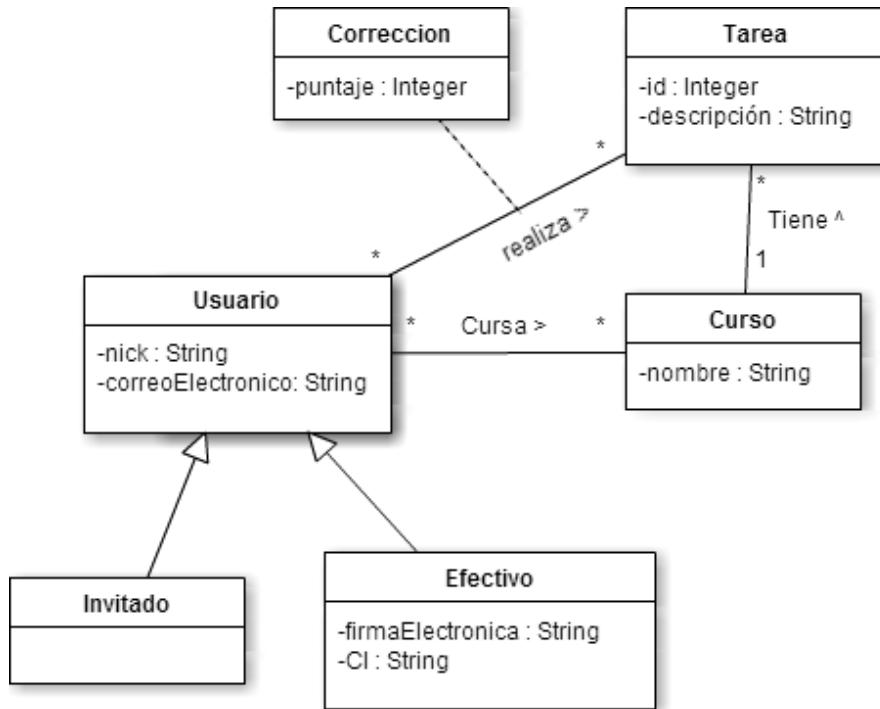
Parte b)

A usted y a su equipo de desarrollo se le ha encomendado la tarea de diseñar un módulo específico de un Sistema que permite la realización de cursos on-line por parte de usuarios.

Básicamente los Usuarios pueden ser de dos tipos: Invitado y Efectivo. Todos los usuarios pueden inscribirse a los cursos (esto es: acceder al material, realizar las Tareas, etc) con la única diferencia de que los usuarios efectivos tendrán diploma y los invitados no. Por ese motivo se desea conocer de los efectivos una firma electrónica y su cédula de identidad.

Los cursos poseen un conjunto de tareas que los usuarios deberán realizar. El Sistema debe mantener registro de los puntajes obtenidos por los usuarios en cada tarea.

A continuación se muestra un modelo de dominio obtenido en la etapa de análisis:



Restricciones

- Los Usuarios se identifican por el nick.
- Las Tareas se identifican por el id dentro del curso únicamente.
- Los Cursos se identifican por el nombre.
- Un Usuario realiza Tareas pertenecientes a un Curso al que está inscripto.

Se consideran los casos de uso “Agregar nueva corrección a Usuario” y “Obtener información de un Curso”, cada uno de los cuales es modelado con una única operación del Sistema, cuyos contratos se especifican a continuación:

agregarNuevaCorreccion (nick, nombreCurso : String, idTarea, puntaje: Integer)	
Descripción	Se agrega un nueva Corrección al Usuario nick en una Tarea realizada por él.
Parámetros	<ul style="list-style-type: none"> - nick: Nick del Usuario. - nombreCurso: nombre del Curso. - idTarea: Id de la Tarea a la que se le asignará la corrección. - puntaje: Puntaje obtenido por el Usuario en la Tarea.
Pre-condiciones	<ul style="list-style-type: none"> - Existe en el Sistema una instancia de Usuario con nick igual a nick. - Existe en el Sistema una instancia de Curso con nombre igual a nombreCurso. - Existe en el Sistema una instancia de Tarea con id igual a idTarea asociada a un Curso con nombre igual a nombreCurso. - NO existe una instancia de tipo asociativo Corrección entre un Usuario con nick igual a nick, y una Tarea cuyo id es igual a idTarea asociada a un Curso con nombre igual a nombreCurso. - Existe un link entre una instancia de Usuario cuyo nick es igual a

	nick y una instancia de Curso cuyo nombre es igual a nombreCurso.
Post- condiciones	- Existe en el Sistema una nueva instancia de tipo asociativo Corrección con puntaje igual a puntaje, entre la instancia de Curso cuyo nombre es igual a nombreCurso y la instancia de Usuario cuyo Nick es igual a nick

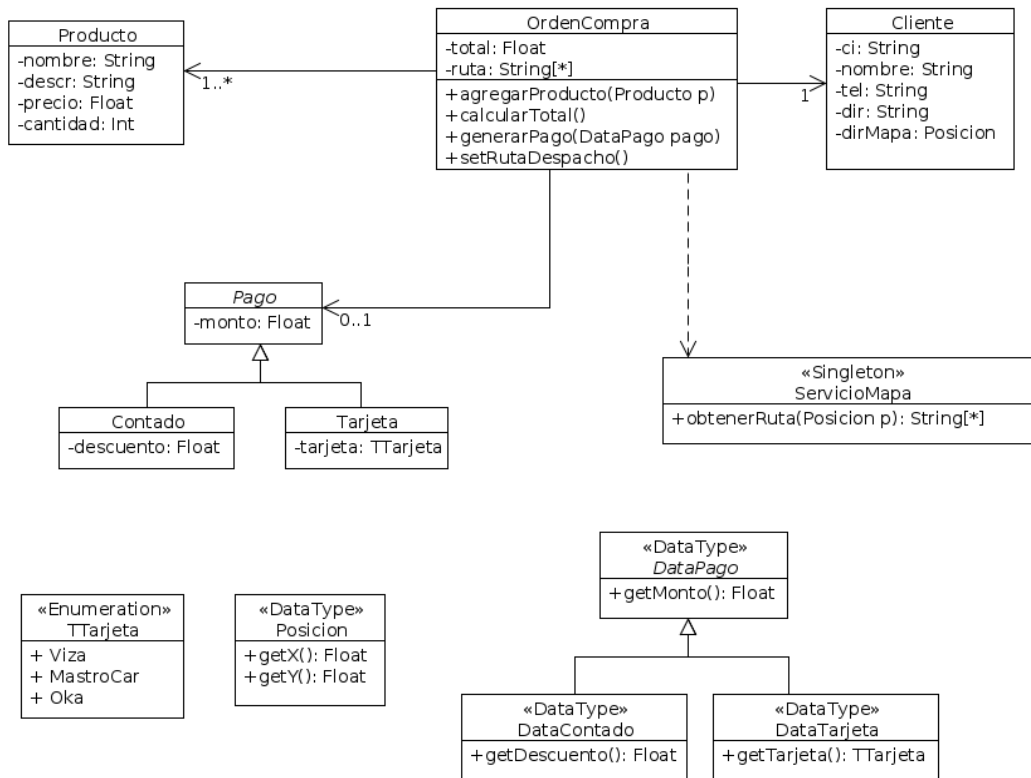
obtenerInformaciónDeCurso(nombreCurso : String)	
Descripción	Se obtiene información de un Curso
Parámetros	- nombreCurso: nombre del Curso del que se desea obtener información.
Pre- condiciones	- Existe en el Sistema una instancia de Curso con nombre igual a nombreCurso.
Post- condiciones	- Se retorna un datavalue con la siguiente información: <ul style="list-style-type: none"> o Nombre del Curso cuyo nombre sea igual a nombreCurso. o Nick y correo electrónico de todos los usuarios inscriptos al curso cuyo nombre es igual a nombreCurso. En caso de que sean Usuarios efectivos se retorna su firma electrónica y su C.I. o Id y descripción de todas las Tareas asociadas al curso cuyo nombre es igual a nombreCurso.

Se pide:

- i. Realizar el Diagrama de Comunicación (incluyendo visibilidades) para cada una de las operaciones previamente declaradas.
- ii. Realizar el Diagrama de Clases de Diseño correspondiente.

Problema 3 (35 puntos)

Su equipo de diseño confeccionó el siguiente diagrama, que corresponde a una porción de un sistema de facturación y pago online.



Se pide:

Implementar completamente en C++ todas las clases y enumerados, con sus respectivas operaciones, que se muestran en el anterior diagrama de clases.

Observaciones:

- La operación `calcularTotal` debe recorrer todos los productos para computar el total de la orden de compra.
- La operación `setRutaDespacho` debe utilizar `ServicioMapa` para obtener la ruta dada la posición del `Cliente`.
- No realizar el método de la operación `obtenerRuta` en `ServicioMapa`, pero si definirla en la clase.
- Puede suponer la existencia de la interface `ICollectionable` e implementaciones de `ICollection` (clase `List`) e `IIterator` según sea necesario.
- Es posible utilizar las clases `set<T>` o `vector<T>` de la STL.
- Las implementaciones deben incluir constructores, destructores
- Asumir existencia y no implementar los `get` y `set` de los atributos.

- No incluir directivas al precompilador.
- No implementar los datatypes.