

Programación 4

PARCIAL FINAL EDICIÓN 2024

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 (30 puntos)

Se desea crear un sistema para gestionar campeonatos de fútbol. Los usuarios son registrados con su email (único), nombre, teléfono y tipo (exclusivo), que puede ser jugador, juez o delegado. Cada campeonato se identifica con un nombre y tiene hasta 20 equipos participantes registrados por su delegado (único) así como al menos 10 jueces quienes arbitrarán en él. Los equipos pueden participar de múltiples campeonatos a la vez. De cada equipo se conocen desde 11 hasta 22 jugadores (además de su delegado), todos exclusivos por equipo. De cada jugador se conoce además su fecha de nacimiento y posición (arquero, defensor, mediocampista, delantero); todos los equipos deben tener al menos un jugador con posición de arquero. Cada campeonato se organiza en fechas numeradas secuencialmente (e.g. 1, 2) donde se cruzan los equipos en partidos hasta competir todos contra todos, una única vez. De los partidos se conoce la fecha (día y hora) así como la ubicación (dirección de la cancha) donde se realiza y los jueces (hasta 3) asignado(s) al encuentro. Para cada partido jugado los delegados de cada equipo registran las estadísticas por jugador considerando: minutos jugados, goles, asistencias, faltas y tarjetas amarillas y rojas en dicho partido.

Además, se cuenta con la descripción del siguiente Caso de Uso:

Nombre	Alta estadística jugador en partido
Actor	Delegado
Descripción	El caso de uso comienza cuando un usuario delegado desea agregar la estadística de un jugador en un partido. Para eso ingresa al sistema indicando su email y teléfono. El sistema lista todos campeonatos en los que su equipo esté participando para que elija en cual desea agregar estadísticas. El delegado elige el campeonato, el sistema lista todas las fechas para las cuales están faltando estadísticas del equipo. El delegado elige la fecha para la cual desea agregar estadísticas. Se listan todos los jugadores para los cuales faltan estadísticas de su equipo. Mientras que el usuario desee seguir agregando estadísticas para esa fecha, indica el jugador y valores de todas las estadísticas para esa fecha. Finalmente, el sistema lista todas las estadísticas ingresadas incluyendo la información del jugador, para que confirme o cancele.

Se pide:

- Realizar el Modelo de Dominio de la realidad planteada, incluyendo todas las restricciones que considere necesarias en lenguaje natural.
- Realizar un Diagrama de Secuencia del Sistema (DSS) para el Caso de Uso. Indique el uso de memoria del Sistema y de datatypes, si corresponde.

Problema 2 (30 puntos)

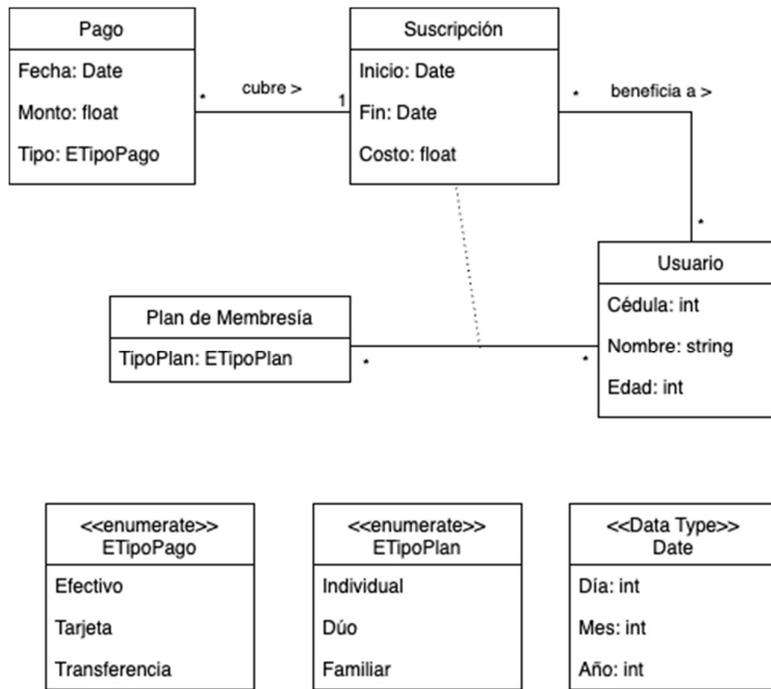
Considere la siguiente realidad para un sistema de gestión de membresías de usuarios.

Los usuarios pueden contratar membresías, las cuales se definen según el tipo de plan: Individual, Dúo y Familiar. Al contratar una membresía, se crea una suscripción que beneficia al usuario contratante y, de acuerdo al tipo de plan, puede beneficiar a usuarios adicionales según el siguiente detalle:

- Plan Individual: beneficia solo al usuario que contrata la membresía.
- Plan Dúo: beneficia al usuario que contrata y opcionalmente a otro usuario.
- Plan Familiar: beneficia al usuario contratante y hasta a otros cuatro usuarios adicionales.

El usuario que decide asociarse a un plan está obligado a realizar pagos para cubrir el costo de la membresía. Los pagos pueden ser parciales, es decir, una suscripción puede ser cubierta mediante varios pagos que, en conjunto, cubren el total del costo. De cada pago se registran sus datos.

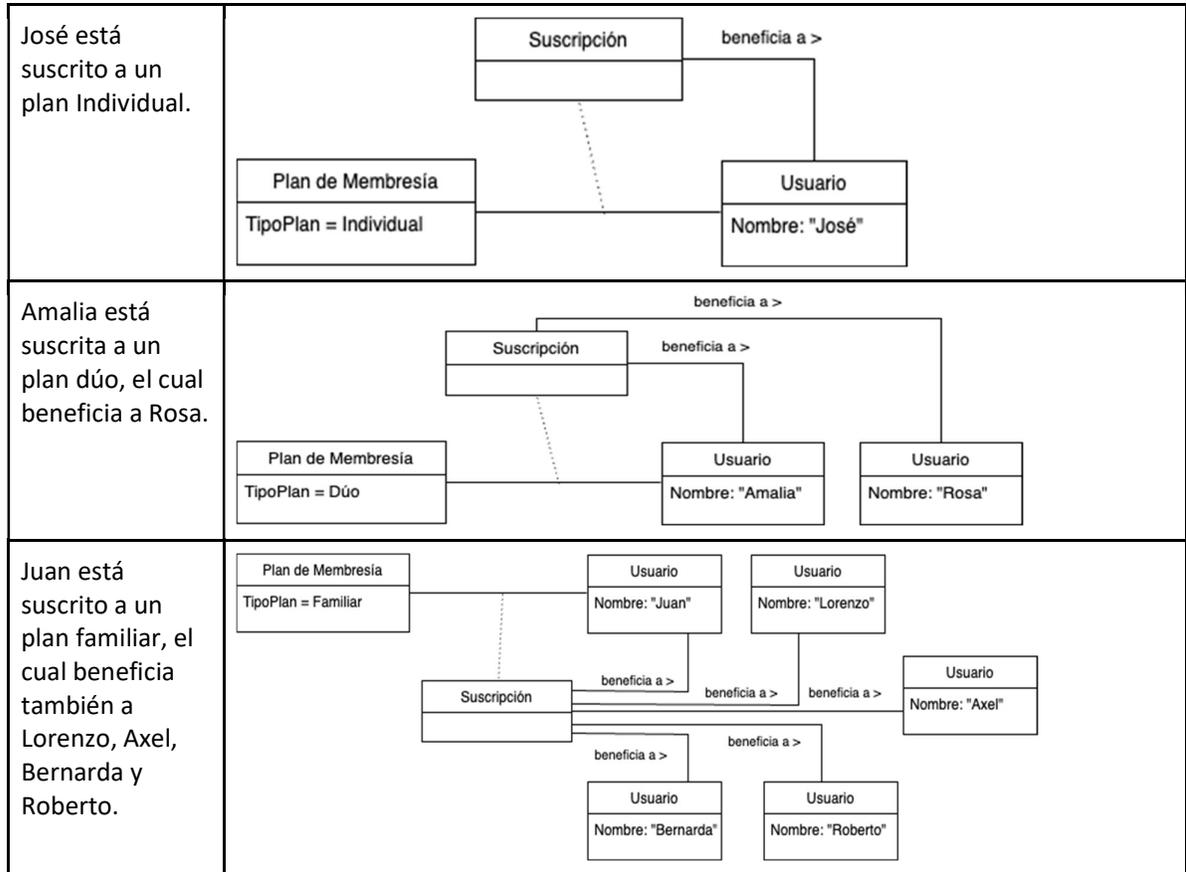
Se realizó el siguiente modelo de dominio de la realidad.



Restricciones:

- Si un Usuario está asociado a través de una Suscripción a un Plan de Membresía de tipo Individual, entonces esa Suscripción beneficia únicamente a ese Usuario.
- Si el Plan es Dúo, la Suscripción beneficia hasta un Usuario además del suscrito.
- Si el Plan es Familiar, la Suscripción beneficia hasta cuatro Usuarios además del suscrito.
- El TipoPlan identifica al Plan de la Membresía
- La suma de todos los montos de los Pagos asociados a una Suscripción es igual a su costo.

A continuación, se presentan ejemplos (en términos del modelo) de suscripciones a cada tipo de plan. La mayoría de los atributos de las instancias no se incluyen para mantener claridad.



Se definieron las siguientes operaciones del sistema:

Nombre	listarFamiliaresPorPago(año: int) : Set(int)
Descripción	Retorna todas las cédulas de identidad de usuarios que sean beneficiarios de una suscripción asociada a un Plan de Membresía Familiar con algún pago realizado en el año recibido por parámetro.
Parámetros	año: Año para el cual se listan los usuarios.
Precondiciones	Existe una instancia de Plan de Membresía con TipoPlan = Familiar.
Poscondiciones	Se devuelve una lista de cédulas de identidad de los usuarios que estén asociados a una suscripción con un Plan de Membresía Familiar y que además tengan un pago asociado a esa suscripción durante el año recibido como parámetro.

Nombre	agregarSuscripcionFamiliar(cedulaSuscrito:int, inicioSuscripción:Date, finSuscripción:Date, costoSuscripción:float, cedulasBeneficiarios:Set(int))
Descripción	Crea una suscripción a un Plan Familiar.

Parámetros	<p>cedulaSuscrito: Número de cédula de identidad del usuario que tiene la suscripción.</p> <p>inicioSuscripción: Fecha de inicio de la suscripción.</p> <p>finSuscripción: Fecha de fin de la suscripción.</p> <p>costoSuscripción: Costo de la suscripción.</p> <p>cedulasBeneficiarios: Conjunto de los usuarios que se benefician de la suscripción además del usuario suscrito.</p>
Precondiciones	<p>El conjunto {cedulasBeneficiarios U cedulaSuscrito} tiene como máximo cardinalidad 5.</p> <p>Existe una instancia de Plan de Membresía con TipoPlan = Familiar.</p> <p>Existen las instancias de Usuario cuyas cédulas de identidad pertenecen al conjunto {cedulasBeneficiarios U cedulaSuscrito}.</p>
Poscondiciones	<p>Existe una instancia s de Suscripción entre la instancia de Usuario con Cédula=cedulaSuscrito y la instancia de Plan de Membresía con TipoPlan=Familiar, con los valores de atributos Inicio=inicioSuscripción, Fin=finSuscripción, Costo=costoSuscripción.</p> <p>Existe un link entre s y todas las instancias de Usuario con Cédula perteneciente al conjunto {cedulasBeneficiarios U cedulaSuscrito}.</p>

Se pide:

- Realizar los diagramas de comunicación correspondientes a las dos operaciones especificadas, incluyendo las visibilidades. Asuma que en el sistema hay una colección global de usuarios y otra colección global de Planes de Membresía.
- Realizar el diagrama de clases de diseño (DCD) resultante del diseño realizado en la parte anterior; incluya la clase Sistema.

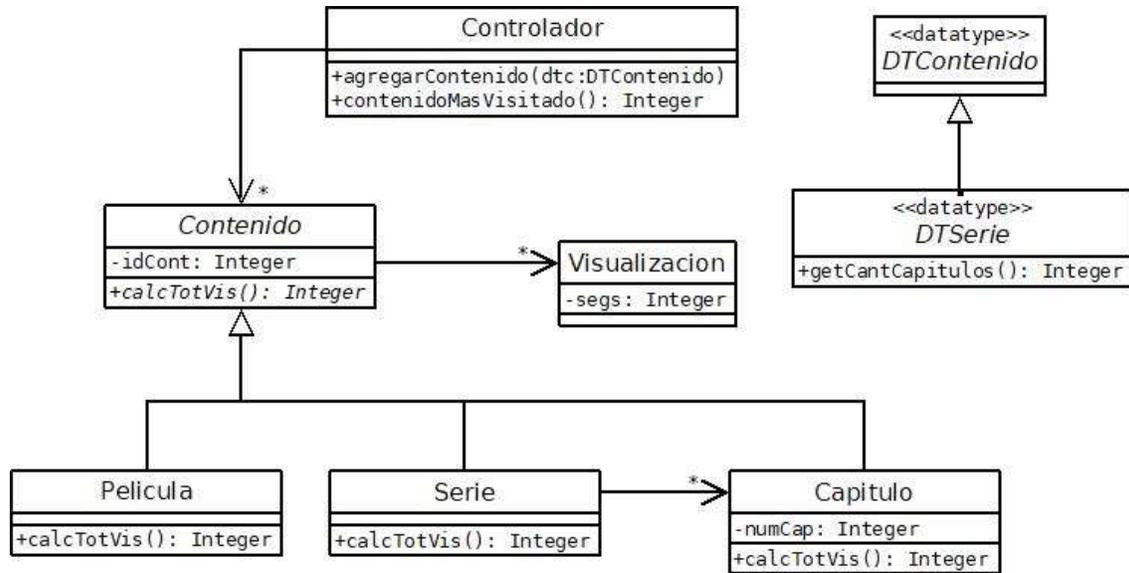
Problema 3 (25 puntos)

Se está desarrollando un componente de cálculo de estadísticas de visualizaciones para una plataforma en línea de películas y series. La figura de la siguiente página muestra el diseño, donde la clase Contenido se especializa en las clases Pelicula, Serie y Capitulo. A su vez, una serie está compuesta por varios capítulos. Por su parte, cada capítulo, además de tener su identificador de contenido, que es un entero positivo correlativo generado por el sistema, tiene un identificador propio, que es otro entero positivo correlativo dentro de la serie. Cada vez que un usuario visualiza una película o un capítulo de serie, se genera una instancia de Visualizacion según la estructura del diagrama, donde el atributo segs registra la cantidad de segundos de la visualización. Para las instancias de Serie no se registran visualizaciones explícitamente, solo para sus capítulos. Por otra parte, la jerarquía de datatypes describe la estructura diseñada para el ingreso de los datos al sistema.

El comportamiento de las operaciones del controlador es el siguiente:

- `agregarContenido`: Da de alta un nuevo contenido en el sistema, en base a los datos del parámetro `dtc`, que solo refiere a películas o series. Si es una serie, además se debe crear cada instancia de capítulo con su correspondiente número, vincularlo a la serie y darlo de alta en el sistema.
- `contenidoMasVisitado`: Devuelve el identificador del contenido con mayor tiempo acumulado de visualización, ya sea una película, serie o capítulo. Para eso utiliza la operación `calcTotVis`, que acumula todas las visualizaciones del contenido. En el caso de una serie, se

debe devolver la acumulación de las visualizaciones de todos sus capítulos. El resultado puede referir a una película, serie o capítulo. Asumir que: (i) al menos existe un contenido en el sistema, (ii) si hay dos contenidos con máximo tiempo de visualización, retornar cualquiera de los dos.

**Se pide:**

Implementar completamente (.h y .cpp) las clases **Controlador**, **Contenido**, **Serie** y **Capítulo**.

Observaciones:

- NO se pueden agregar más operaciones de las definidas en el diseño anterior.
- Utilice colecciones paramétricas (contenedores STL).
- Incluya los constructores que sean necesarios, no incluya destructores.
- No incluya directivas al precompilador (`#include`, etc).
- No es necesario implementar setters y getters adicionales de las clases; puede asumir su existencia si se requieren en algún método.