

Ejercicio 1 (3 puntos)

Se desea desarrollar una aplicación de gestión de noticias. Una noticia está compuesta de título, contenido textual, fecha y hora de publicación, y está asociada a una temática. La aplicación maneja un conjunto de temáticas acotado, cada una de las cuales tiene con un nombre y descripción.

Un usuario de la aplicación tiene asociado un identificador numérico único, y puede solicitar la recepción automática de noticias de última hora. Para ello, puede registrar las temáticas de las noticias sobre las que está interesado, y los tipos de transmisores por los que recibir las noticias: email (a través de su dirección de correo electrónico), SMS (a través de su número de teléfono) o Twitter (a través de su nombre de usuario).

Un usuario puede modificar su registro, añadiendo o eliminando temáticas o transmisores.

Al recibir una noticia de última hora, la aplicación la envía a todos aquellos usuarios con interés en la temática de la noticia. Un envío se realiza mediante todos los transmisores registrados por cada usuario.

Se pide:

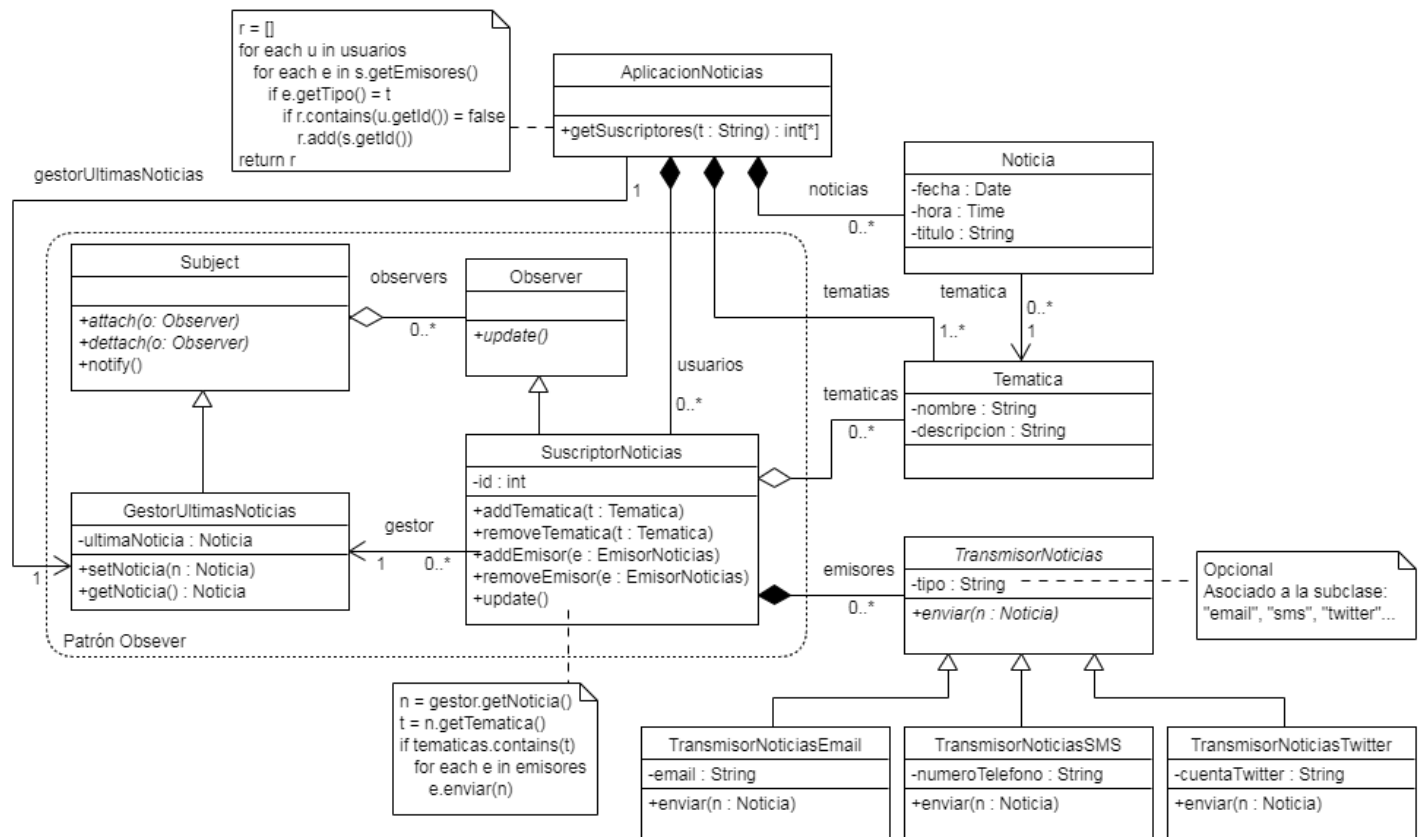
- a. Realiza el diagrama de clases en UML con el diseño de la aplicación descrita arriba. Indica en el diagrama los **patrones de diseño** que uses, incluyendo el rol de los elementos de tu diagrama en dichos patrones. **(2 puntos)**.

En el diagrama no incluyas constructores, getters ni setters, pero sí los métodos necesarios para cumplir los requisitos del enunciado.

- b. Incluye en el diagrama y proporciona el pseudocódigo de los siguientes métodos: **(1 punto)**.

- b.1. Obtener los identificadores de los usuarios que reciben noticias por un tipo de transmisor dado.
b.2. Enviar una noticia de última hora a los usuarios correspondientes.

Entregar la solución del ejercicio en un único fichero, con formato PDF, PNG o JPG, y con nombre “**apellidos_nombre.pdf**” donde “apellidos” y “nombre” han de ser los tuyos, p.e., GarcíaPérez_JoseMaría.pdf. Abre el fichero antes de enviarlo para verificar que es el correcto.

Solución:

(criterios de corrección en la siguiente página)

Criterios de corrección

EC = Evaluación Continua; ENC = Evaluación No Continua

- Elementos generales del diagrama (0.25 puntos en EC; 0.2 puntos en ENC).
 - Uso correcto del estándar UML: símbolos para clases, enumeraciones, interfaces, comentarios y relaciones (agregaciones, asociaciones); formatos para nombre y tipo de atributos (incluidos arrays []) y prototipos de métodos.
 - Uso de nombres adecuados para clases y atributos: significados interpretables, singulares vs. plurales, mayúsculas vs. minúsculas.
- Relaciones (0.25 puntos en EC; 0.2 en ENC).
 - Existencia/falta de relaciones relevantes.
 - Existencia de relaciones de agregación y contenido. Se ha dado por buena alguna variante a la de la solución, como por ejemplo una relación con cardinalidades en el patrón Observer, y una agregación (rombo blanco) entre Noticia-ElementoNoticia y Oferta-Evaluacion (ambas del modelo de examen de evaluación no continua).
 - Nombrado de asociaciones.
 - Cardinalidad correcta en las relaciones.
 - No “duplicado” de datos: atributo y relación que reflejan los mismos datos.
- Patrón Observer (0.5 puntos en EC; 0.5 en ENC).
 - Identificación del patrón.
 - Asignación correcta de los roles del patrón a las clases correspondientes.
 - Adición de los métodos del patrón en las clases correspondientes.
- Clase Usuario/Suscriptor (0.5 puntos en EC; 0.4 en ENC).
 - Adición de los atributos/relaciones y métodos asociados a los requisitos del enunciado: añadir/eliminar categorías de interés, añadir/eliminar emisores de información.
- Clase Emisor (0.5 puntos en EC; 0.5 en ENC).
 - Consideración de subclases para los diferentes tipos de emisor, y definición de la clase Emisor como abstracta.
 - Adición de los atributos y método “enviar” a la clase Emisor y subclases.
- Clase Categoría (0.3 puntos en ENC).
 - Modelado correcto de la taxonomía categoría-subcategorías.
- Clases adicionales: contenidos de noticias y evaluaciones de ofertas (0.3 puntos en ENC).
 - Modelado correcto de clases (incluyendo sus atributos) cumpliendo las restricciones adicionales del enunciado de ENC.
- Prototipo del método del apartado b.1 (0.2 puntos en EC; 0.1 en ENC).
- Pseudocódigo del método del apartado b.1 (0.3 puntos en EC; 0.2 en ENC).
- Prototipo del método del apartado b.2 (0.2 puntos en EC; 0.1 en ENC).
- Pseudocódigo del método del apartado b.2 (0.3 puntos en EC; 0.2 en ENC).