Programación 2

Tecnicatura en Desarrollo de Aplicaciones Informáticas

Recuperatorio 2022

Punto a)

Modelar un sistema que permita la organización de noticias de un portal electrónico. De cada noticia se guarda: el contenido (texto), el título, el autor, la categoría (“policial”, “espectáculos”, etc, solo tiene una única categoría y la misma no puede modificarse una vez creada la noticia) y una lista de palabras claves. Las noticias del portal se organizan en una estructura de secciones, subsecciones, grupos y subgrupos. Una noticia puede estar contenida dentro un subgrupo, grupo, subsección o sección. Cada sección/subsección/ grupo/subgrupo posee su propia categoría que no se permite |cambiar una vez creada. Por otro lado, las palabras claves de una sección/subsección/ grupo/subgrupo, se calculan como la unión de todas las palabras claves de los elementos que contiene, sin incluir palabras repetidas. Todos los elementos dentro de una sección/subsección/grupo/subgrupo, se encuentran ordenados por categoría de forma alfabética ascendente. Cada vez que se le agregue un elemento (no importa del tipo que sea) **el orden se debe mantener**.

Punto a)

Modelar un sistema que permita la organización de noticias de un portal electrónico. De cada **noticia** se guarda: el contenido (texto), el título, el autor, la categoría (“policial”, “espectáculos”, etc, solo tiene una única categoría y la misma no puede modificarse una vez creada la noticia) y una lista de palabras claves. Las noticias del portal se organizan en una estructura de secciones, subsecciones, grupos y subgrupos. Una noticia puede estar contenida dentro un subgrupo, grupo, subsección o sección. Cada sección/subsección/ grupo/subgrupo posee su propia categoría que no se permite |cambiar una vez creada. Por otro lado, las palabras claves de una sección/subsección/ grupo/subgrupo, se calculan como la unión de todas las palabras claves de los elementos que contiene, sin incluir palabras repetidas. Todos los elementos dentro de una sección/subsección/grupo/subgrupo, se encuentran ordenados por categoría de forma alfabética ascendente. Cada vez que se le agregue un elemento (no importa del tipo que sea) **el orden se debe mantener**.

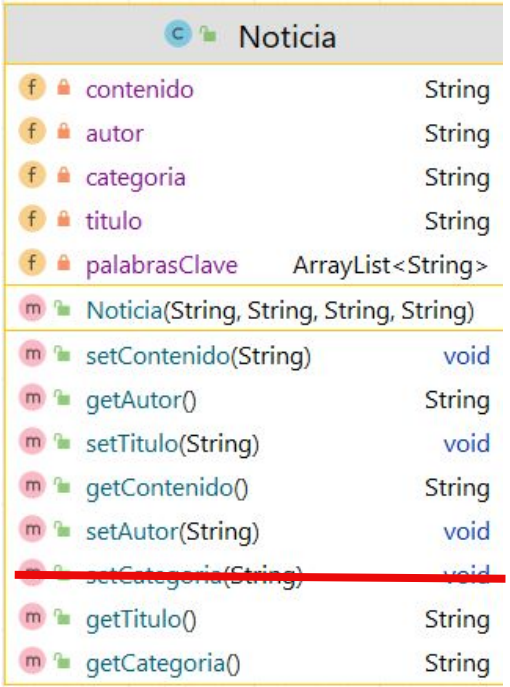
Punto a)

Modelar un sistema que permita la organización de noticias de un portal electrónico. De cada **noticia** se guarda: el contenido (texto), el título, el autor, la categoría (“policial”, “espectáculos”, etc, solo tiene una única categoría y la misma no puede modificarse una vez creada la noticia) y una lista de palabras claves. Las noticias del portal se organizan en una estructura de secciones, subsecciones, grupos y subgrupos. Una noticia puede estar contenida dentro un subgrupo, grupo, subsección o sección. Cada sección/subsección/ grupo/subgrupo posee su propia categoría que no se permite |cambiar una vez creada. Por otro lado, las palabras claves de una sección/subsección/ grupo/subgrupo, se calculan como la unión de todas las palabras claves de los elementos que contiene, sin incluir palabras repetidas. Todos los elementos dentro de una sección/subsección/grupo/subgrupo, se encuentran ordenados por categoría de forma alfabética ascendente. Cada vez que se le agregue un elemento (no importa del tipo que sea) **el orden se debe mantener**.

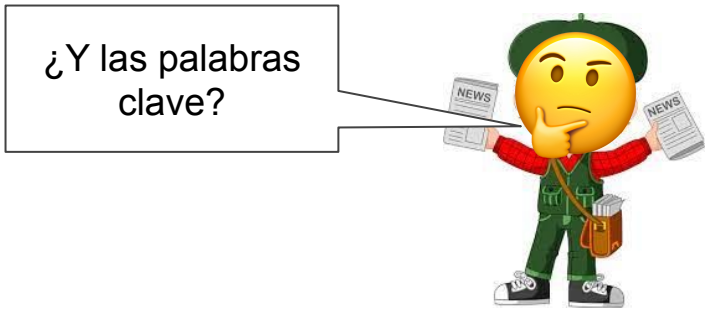
Noticia	
contenido	String
autor	String
categoria	String
titulo	String
palabrasClave	ArrayList<String>
Noticia(String, String, String, String)	
setContenido(String)	void
getAutor()	String
setTitulo(String)	void
getContenido()	String
setAutor(String)	void
setCategoria(String)	void
getTitulo()	String
getCategoria()	String

Punto a)

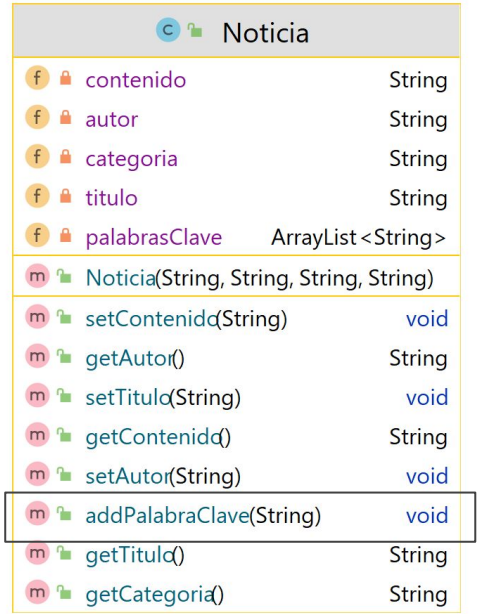
Modelar un sistema que permita la organización de noticias de un portal electrónico. De cada **noticia** se guarda: el contenido (texto), el título, el autor, la categoría (“policial”, “espectáculos”, etc, solo tiene una única categoría y la misma no puede modificarse una vez creada la noticia) y una lista de palabras claves. Las noticias del portal se organizan en una estructura de secciones, subsecciones, grupos y subgrupos. Una noticia puede estar contenida dentro un subgrupo, grupo, subsección o sección. Cada sección/subsección/ grupo/subgrupo posee su propia categoría que no se permite |cambiar una vez creada. Por otro lado, las palabras claves de una sección/subsección/ grupo/subgrupo, se calculan como la unión de todas las palabras claves de los elementos que contiene, sin incluir palabras repetidas. Todos los elementos dentro de una sección/subsección/grupo/subgrupo, se encuentran ordenados por categoría de forma alfabética ascendente. Cada vez que se le agregue un elemento (no importa del tipo que sea) **el orden se debe mantener**.



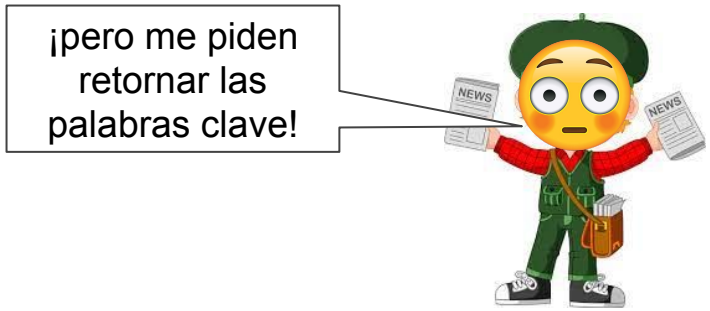
Punto a)



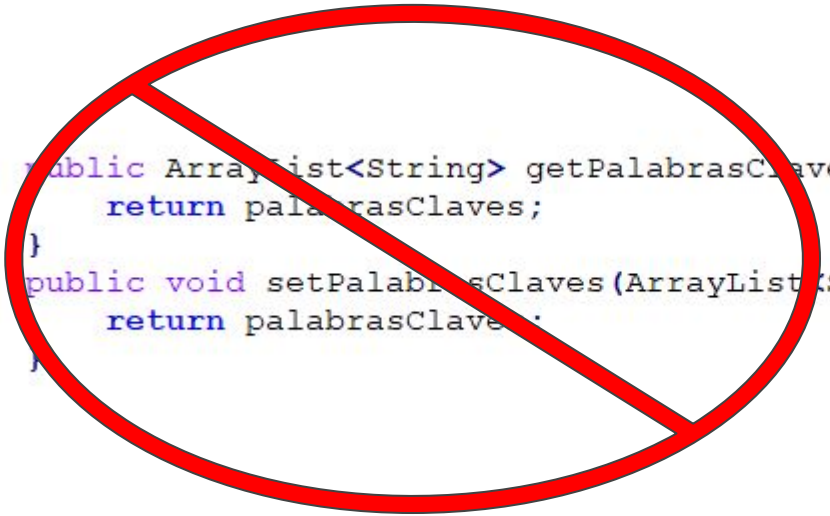
Modelar un sistema que permita la organización de noticias de un portal electrónico. De cada **noticia** se guarda: el contenido (texto), el título, el autor, la categoría (“policial”, “espectáculos”, etc, solo tiene una única categoría y la misma no puede modificarse una vez creada la noticia) y una lista de palabras claves. Las noticias del portal se organizan en una estructura de secciones, subsecciones, grupos y subgrupos. Una noticia puede estar contenida dentro un subgrupo, grupo, subsección o sección. Cada sección/subsección/ grupo/subgrupo posee su propia categoría que no se permite |cambiar una vez creada. Por otro lado, las palabras claves de una sección/subsección/ grupo/subgrupo, se calculan como la unión de todas las palabras claves de los elementos que contiene, sin incluir palabras repetidas. Todos los elementos dentro de una sección/subsección/grupo/subgrupo, se encuentran ordenados por categoría de forma alfabética ascendente. Cada vez que se le agregue un elemento (no importa del tipo que sea) **el orden se debe mantener**.



Manejo de Arraylist palabrasClaves



- No hay que romper el encapsulamiento!
- El ArrayList se crea en el constructor
  - **Nadie fuera de la clase lo conoce**



ROTURA DE ENCAPSL

Noticia	
contenido	String
autor	String
categoria	String
titulo	String
palabrasClave	ArrayList<String>
Noticia(String, String, String, String)	
setContenido(String)	void
getAutor()	String
setTitulo(String)	void
getContenido()	String
setAutor(String)	void
addPalabraClave(String)	void
getTitulo()	String
getCategoria()	String

Cualquiera desde afuera puede manipular el ArrayList sin pasar por la clase

Manejo de Arraylist palabrasClaves

¡pero me piden retornar las palabras clave!

- No hay que romper el encapsulamiento!
- El ArrayList se crea en el constructor
  - **Nadie fuera de la clase lo conoce**

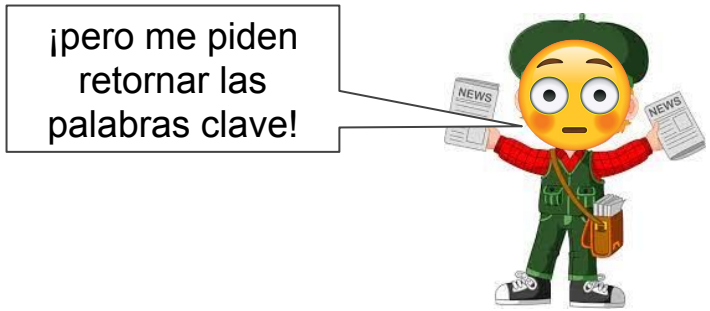
Devolver el ArrayList

```
public ArrayList<String> getPalabrasClaves(){
    return new ArrayList<String>(palabrasClaves);
}

public ArrayList<String> getPalabrasClaves(){
    ArrayList<String> aux = new ArrayList<String>();
    aux.addAll(palabrasClaves);
    return aux;
}

public ArrayList<String> getPalabrasClaves(){
    ArrayList<String> aux = palabrasClaves;
    return aux;
}
```

Crea un Arreglo nuevo y le agrega las palabras claves

Crea una Variable nueva y “APUNTA” al mismo arreglo

Manejo de Arraylist palabrasClaves



- No hay que romper el encapsulamiento!
- El ArrayList se crea en el constructor
  - **Nadie fuera de la clase lo conoce**

Agregar al ArrayList

```
public void addPalabrasClaves(String s){
  if (!palabrasClaves.contains(s) ){
     palabrasClaves.add(s)
  }
}
```

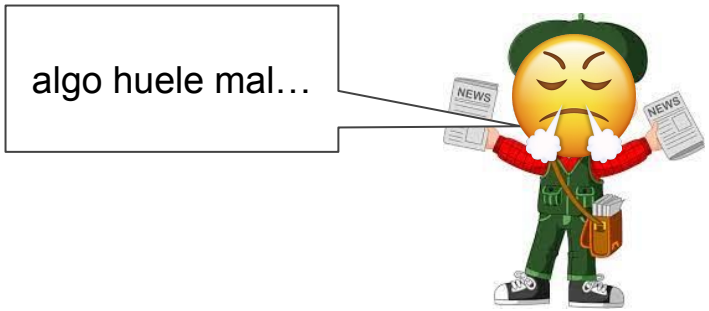
+ Controlo repetidos
+ No necesito implementar "equals" porque se usa el de String

Noticia	
contenido	String
autor	String
categoria	String
titulo	String
palabrasClave	ArrayList<String>
Noticia(String, String, String, String)	
setContenido(String)	void
getAutor()	String
setTitulo(String)	void
getContenido()	String
setAutor(String)	void
addPalabraClave(String)	void
getPalabrasClave()	ArrayList<String>
getTitulo()	String
getCategoria()	String

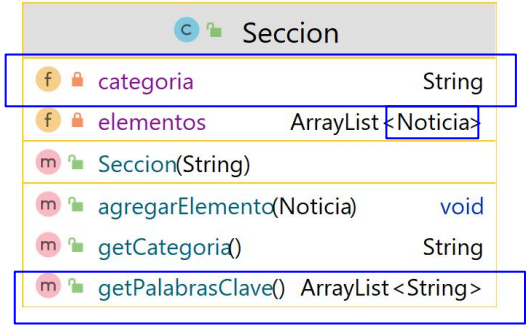
Punto a)

Modelar un sistema que permita la organización de noticias de un portal electrónico. De cada **noticia** se guarda: el contenido (texto), el título, el autor, la categoría (“policial”, “espectáculos”, etc, solo tiene una única categoría y la misma no puede modificarse una vez creada la noticia) y una lista de palabras claves. Las noticias del portal se organizan en una estructura de secciones, subsecciones, grupos y subgrupos. Una noticia puede estar contenida dentro un subgrupo, grupo, subsección o sección. Cada sección/subsección/ grupo/subgrupo posee su propia categoría que no se permite cambiar una vez creada. Por otro lado, las palabras claves de una sección/subsección/ grupo/subgrupo, se calculan como la unión de todas las palabras claves de los elementos que contiene, sin incluir palabras repetidas. Todos los elementos dentro de una sección/subsección/grupo/subgrupo, se encuentran ordenados por categoría de forma alfabética ascendente. Cada vez que se le agregue un elemento (no importa del tipo que sea) **el orden se debe mantener**.

Punto a)



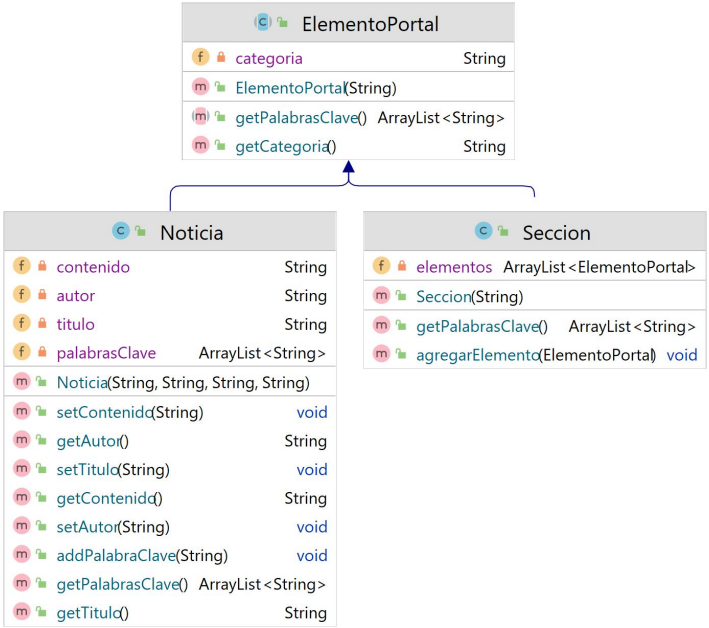
Modelar un sistema que permita la organización de noticias de un portal electrónico. De cada **noticia** se guarda: el contenido (texto), el título, el autor, la categoría (“policial”, “espectáculos”, etc, solo tiene una única categoría y la misma no puede modificarse una vez creada la noticia) y una lista de palabras claves. Las noticias del portal se organizan en una estructura de secciones, subsecciones, grupos y subgrupos. Una noticia puede estar contenida dentro un subgrupo, grupo, subsección o sección. Cada sección/subsección/ grupo/subgrupo posee su propia categoría que no se permite cambiar una vez creada. Por otro lado, las palabras claves de una sección/subsección/ grupo/subgrupo, se calculan como la unión de todas las palabras claves de los elementos que contiene, sin incluir palabras repetidas. Todos los elementos dentro de una sección/subsección/grupo/subgrupo, se encuentran ordenados por categoría de forma alfabética ascendente. Cada vez que se le agregue un elemento (no importa del tipo que sea) **el orden se debe mantener**.



Punto a)

¡Composite!

Modelar un sistema que permita la organización de noticias de un portal electrónico. De cada **noticia** se guarda: el contenido (texto), el título, el autor, la categoría (“policial”, “espectáculos”, etc, solo tiene una única categoría y la misma no puede modificarse una vez creada la noticia) y una lista de palabras claves. Las noticias del portal se organizan en una estructura de secciones, subsecciones, grupos y subgrupos. Una noticia puede estar contenida dentro un subgrupo, grupo, subsección o sección. Cada sección/subsección/ grupo/subgrupo posee su propia categoría que no se permite cambiar una vez creada. Por otro lado, las palabras claves de una sección/subsección/ grupo/subgrupo, se calculan como la unión de todas las palabras claves de los elementos que contiene, sin incluir palabras repetidas. Todos los elementos dentro de una sección/subsección/grupo/subgrupo, se encuentran ordenados por categoría de forma alfabética ascendente. Cada vez que se le agregue un elemento (no importa del tipo que sea) **el orden se debe mantener**.

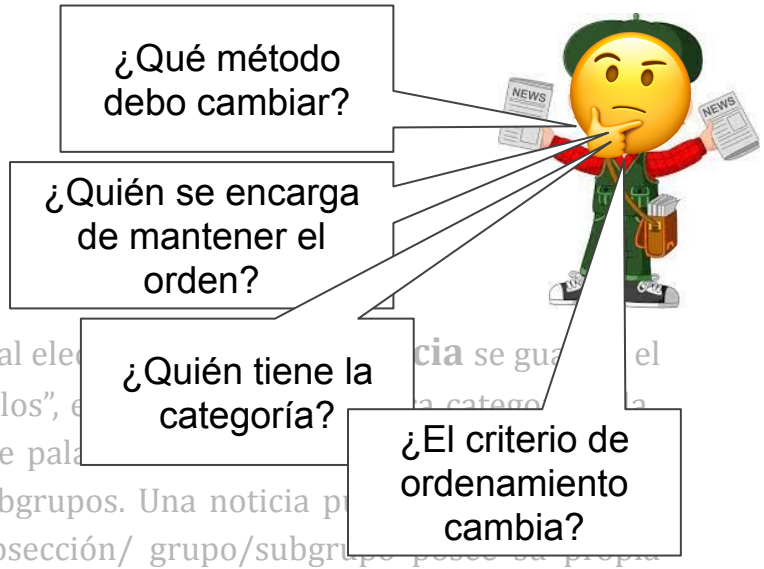


Punto a)

Modelar un sistema que permita la organización de noticia
electrónico. De cada **noticia** se guarda: el contenido (text
autor, la categoría (“policial”, “espectáculos”, etc, solo tie
categoría y la misma no puede modificarse una vez cread
una lista de palabras claves. Las noticias del portal se orga
estructura de secciones, subsecciones, grupos y subgrupo
puede estar contenida dentro un subgrupo, grupo, subsecc
Cada sección/subsección/ grupo/subgrupo posee su pro
que no se permite cambiar una vez creada. Por otro lado, las palabras claves de una sección/subsección/ grupo/subgrupo, se calculan como la unión de todas las palabras claves de los elementos que contiene, sin incluir palabras repetidas. Todos los elementos dentro de una sección/subsección/grupo/subgrupo, se encuentran ordenados por categoría de forma alfabética ascendente. Cada vez que se le agregue un elemento (no importa del tipo que sea) **el orden se debe mantener**.

```
public ArrayList<String> getPalabrasClave(){
    ArrayList<String> unionPalabras = new ArrayList<>();
    for (ElementoPortal hijo:this.elementos){
        ArrayList<String> palabrasHijo = hijo.getPalabrasClave();
        for (String palabra:palabrasHijo)
            if (!unionPalabras.contains(palabra))
                unionPalabras.add(palabra);
    }
    return unionPalabras;
}
```

Punto a)



Modelar un sistema que permita la organización de noticias de un portal ele **cia** se gua el contenido (texto), el título, el autor, la categoría (“policial”, “espectáculos”, e misma no puede modificarse una vez creada la noticia) y una lista de pala organizan en una estructura de secciones, subsecciones, grupos y subgrupos. Una noticia p dentro un subgrupo, grupo, subsección o sección. Cada sección/subsección/ grupo/subgru categoría que no se permite cambiar una vez creada. Por otro lado, las palabras claves de una sección/subsección/ grupo/subgrupo, se calculan como la unión de todas las palabras claves de los elementos que contiene, sin incluir palabras repetidas. Todos los elementos dentro de una sección/subsección/grupo/subgrupo, se encuentran ordenados por categoría de forma alfabética ascendente. Cada vez que se le agregue un elemento (no importa del tipo que sea) **el orden se debe mantener**.

Punto a)

Modelar un sistema que permita la organización de n
electrónico. De cada **noticia** se guarda: el contenid
autor, la categoría (“policial”, “espectáculos”, etc, so
categoría y la misma no puede modificarse una vez
una lista de palabras claves. Las noticias del portal se organizan en una
estructura de secciones, su
puede estar contenida dentro un subgrupo, grupo, subsección o sección.
Cada sección/subsección/ grupo/subgru
que no se permite cambiar una vez creac
claves de una sección/subsección/ grupo/
unión de todas las palabras claves de los
incluir palabras repetidas. Todos los elementos dentro de una sección/subsección/grupo/subgrupo, se encuentran ordenados por categoría de forma alfabética ascendente. Cada vez que se le agregue un elemento (no importa del tipo que sea) **el orden se debe mantener**.

```
public void agregarElemento(ElementoPortal nueva){
    this.elementos.add(nueva);
    Collections.sort(elementos);
}
```

```
public abstract class ElementoPortal implements Comparable<ElementoPortal>
```

```
@Override
public int compareTo(ElementoPortal otro) {
    return this.getCategoria().compareTo(otro.getCategoria());
}
```

Punto a)

Idealmente, no debería reordenar toda la lista cada vez, sino insertar ordenado…


```
public void agregarElemento(ElementoPortal nueva){
    int pos=0;
    while (pos<this.elementos.size() &&
            this.elementos.get(pos).compareTo(nueva)<0)
        pos++;
    this.elementos.add(pos, nueva);
}
```

Modelar un sistema que permita la orga
electrónico. De cada **noticia** se guarda
autor, la categoría ("policial", "espectá
categoría y la misma no puede modific
una lista de palabras claves. Las noticia
estructura de secciones, subsecciones,
puede estar contenida dentro un subgr
Cada sección/subsección/ grupo/subg
que no se permite cambiar una vez creada. Por otro lado, las palabras claves de una sección/subsección/ grupo/subgrupo, se calculan como la unión de todas las palabras claves de los elementos que contiene, sin incluir palabras repetidas. Todos los elementos dentro de una sección/subsección/grupo/subgrupo, se encuentran ordenados por categoría de forma alfabética ascendente. Cada vez que se le agregue un elemento (no importa del tipo que sea) **el orden se debe mantener**.

Punto a)

¿podía usar Comparator?

¿Qué problema tiene?


```java
public class ComparadorCategoria implements Comparator<ElementoPortal> {
    @Override
    public int compare(ElementoPortal o1, ElementoPortal o2) {
        return o1.getCategoria().compareTo(o2.getCategoria());
    }
}
```

```java
public class Seccion extends ElementoPortal{
    protected ArrayList<ElementoPortal> elementos;
    private Comparator comparador;

    public void agregarElemento(ElementoPortal nueva){
        elementos.add(nueva);
        Collections.sort(elementos, comparador);
    }
  ...
```

Cada sección podría tener su propio comparador y su propia forma de ordenar los elementos (NO es lo que se pedía)

Punto a)

— — —

Otra alternativa…

¿Qué problema tiene?


```java
public class ComparadorCategoria implements Comparator<ElementoPortal> {
    @Override
    public int compare(ElementoPortal o1, ElementoPortal o2) {
        return o1.getCategoria().compareTo(o2.getCategoria());
    }
}
```

```java
public void agregarElemento(ElementoPortal nueva, Comparator comparador){
    elementos.add(nueva);
    Collections.sort(elementos, comparador);
}
```

¡PEOR! Cada vez que agrego un elemento podría cambiar el orden de los elementos

…

Punto b)

El portal ofrece un servicio de **restricción temática** para sus lectores. De esta forma, se debe poder obtener una copia de la estructura de noticias respetando la misma organización en secciones, subsecciones, grupos y subgrupos, pero restringida según diferentes criterios, por ejemplo:

- Que contenga solo las noticias cuyo título contenga la palabra “Tandil”
- Que contenga solo las noticias que en el contenido de la misma se encuentre la frase “en 1930”
- Que contenga solo las noticias de la categoría “Sociales” y cuyo autor NO sea “Ricardo Ruben”
- Que contenga solo las noticias que NO tengan la palabra clave “Qatar 2022”
- Cualquier combinación lógica de los anteriores

Tener en cuenta que, si una sección, subsección, grupo o subgrupo, no contiene ninguna noticia que cumpla con el criterio de restricción, la/el misma/o no se debe incluir en la copia. Considerar también que tanto las noticias como las secciones, subsecciones, grupos y subgrupos retornados por este servicio deben ser una copia del original, de tal forma que, si se modifican, el original se mantenga intacto.

Punto b)

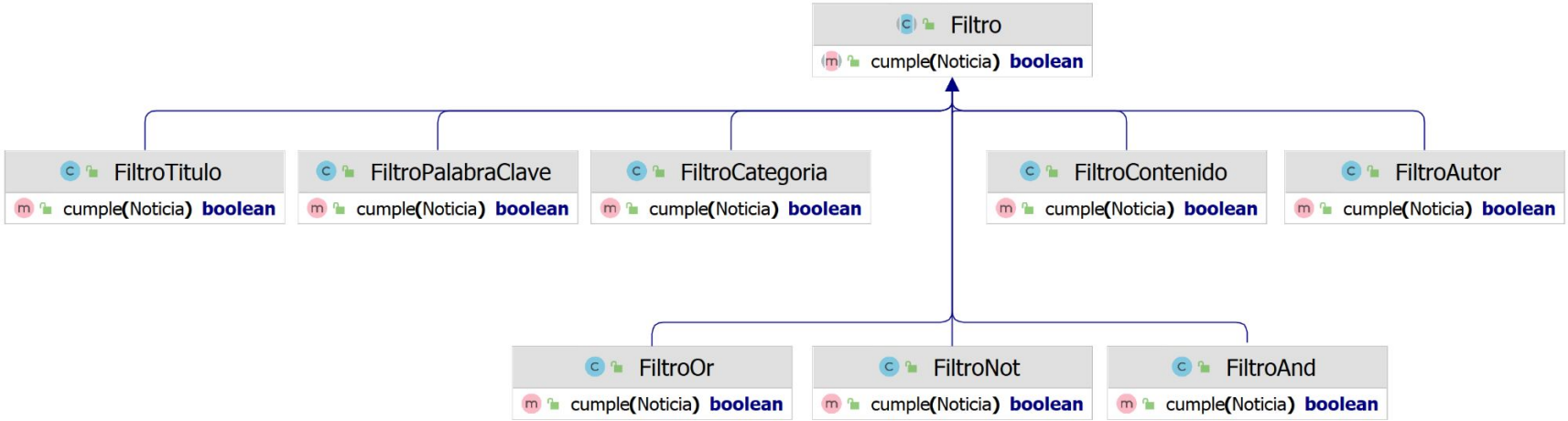
El portal ofrece un servicio de **restricción temática** para sus lectores. De esta forma, se debe poder obtener una copia de la estructura de noticias respetando la misma organización en secciones, subsecciones, grupos y subgrupos, pero restringida según diferentes criterios, por ejemplo:

- Que contenga solo las noticias cuyo título contenga la palabra “Tandil”
- Que contenga solo las noticias que en el contenido de la misma se encuentre la frase “en 1930”
- Que contenga solo las noticias de la categoría “Sociales” y cuyo autor NO sea “Ricardo Ruben”
- Que contenga solo las noticias que NO tengan la palabra clave “Qatar 2022”
- Cualquier combinación lógica de los anteriores

Tener en cuenta que, si una sección, subsección, grupo o subgrupo, no contiene ninguna noticia que cumpla con el criterio de restricción, la/el misma/o no se debe incluir en la copia. Considerar también que tanto las noticias como las secciones, subsecciones, grupos y subgrupos retornados por este servicio deben ser una copia del original, de tal forma que, si se modifican, el original se mantenga intacto.

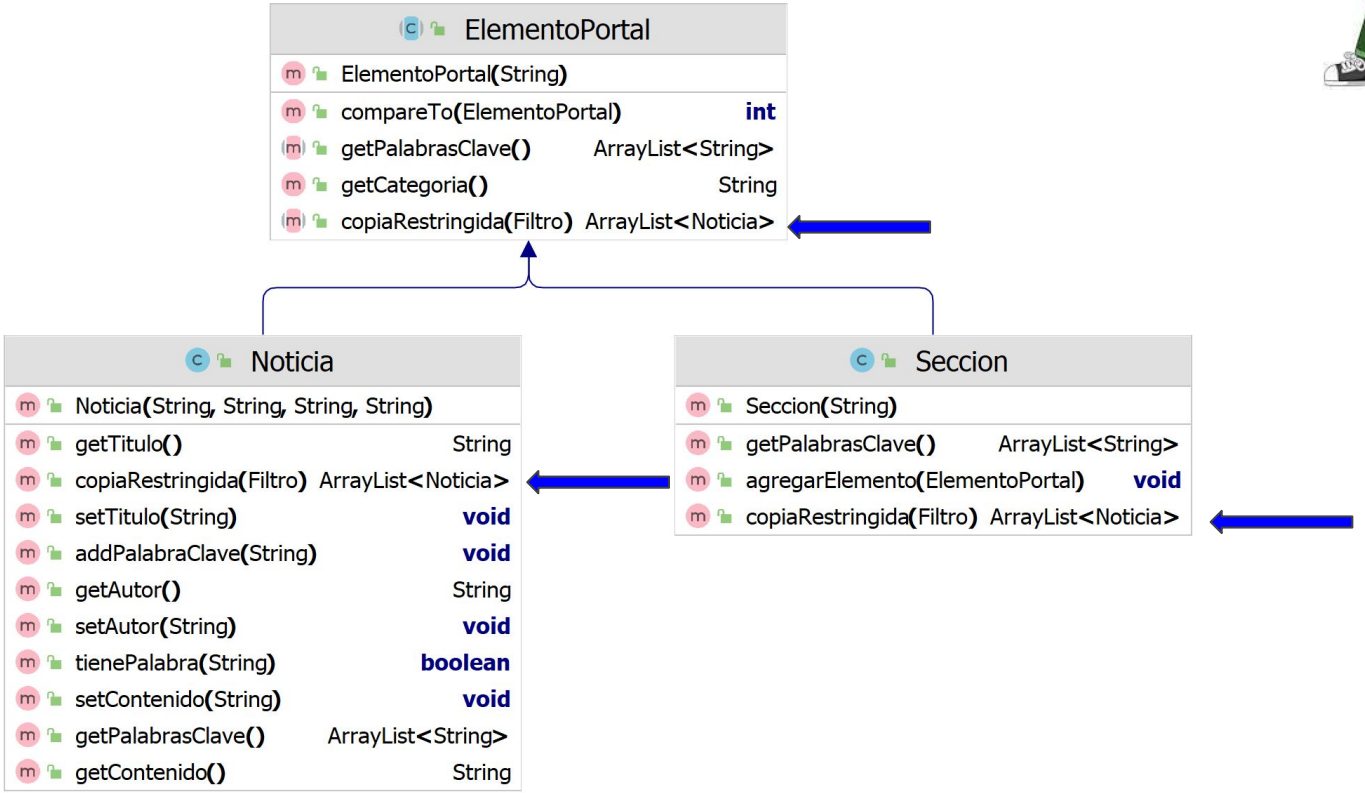
Punto b)


¡Filtros!
eso lo estudié!
NEWS
NEWS


Filtro
cumple(Noticia) boolean
FiltroTitulo
cumple(Noticia) boolean
FiltroPalabraClave
cumple(Noticia) boolean
FiltroCategoria
cumple(Noticia) boolean
FiltroContenido
cumple(Noticia) boolean
FiltroAutor
cumple(Noticia) boolean
FiltroOr
cumple(Noticia) boolean
FiltroNot
cumple(Noticia) boolean
FiltroAnd
cumple(Noticia) boolean

Punto b)

¿Leí bien?



Punto b)

una **copia** de la estructura de noticias **respetando la misma organización** en secciones, subsecciones, grupos y subgrupos

El portal ofrece un servicio de **restricción temática** para sus lectores. De esta forma, se debe poder obtener una copia de la estructura de noticias respetando la misma organización en secciones, subsecciones, grupos y subgrupos, pero restringida según diferentes criterios, por ejemplo:

- Que contenga solo las noticias cuyo título contenga la palabra “Tandil”
- Que contenga solo las noticias que en el contenido de la misma se encuentre la frase “en 1930”
- Que contenga solo las noticias de la categoría “Sociales” y cuyo autor NO sea “Ricardo Ruben”
- Que contenga solo las noticias que NO tengan la palabra clave “Qatar 2022”
- Cualquier combinación lógica de los anteriores

Tener en cuenta que, si una sección, subsección, grupo o subgrupo, no contiene ninguna noticia que cumpla con el criterio de restricción, la/el misma/o no se debe incluir en la copia. Considerar también que tanto las noticias como las secciones, subsecciones, grupos y subgrupos retornados por este servicio deben ser una copia del original, de tal forma que, si se modifican, el original se mantenga intacto.

Punto b)

¡No me están pidiendo el “buscar” clásico!

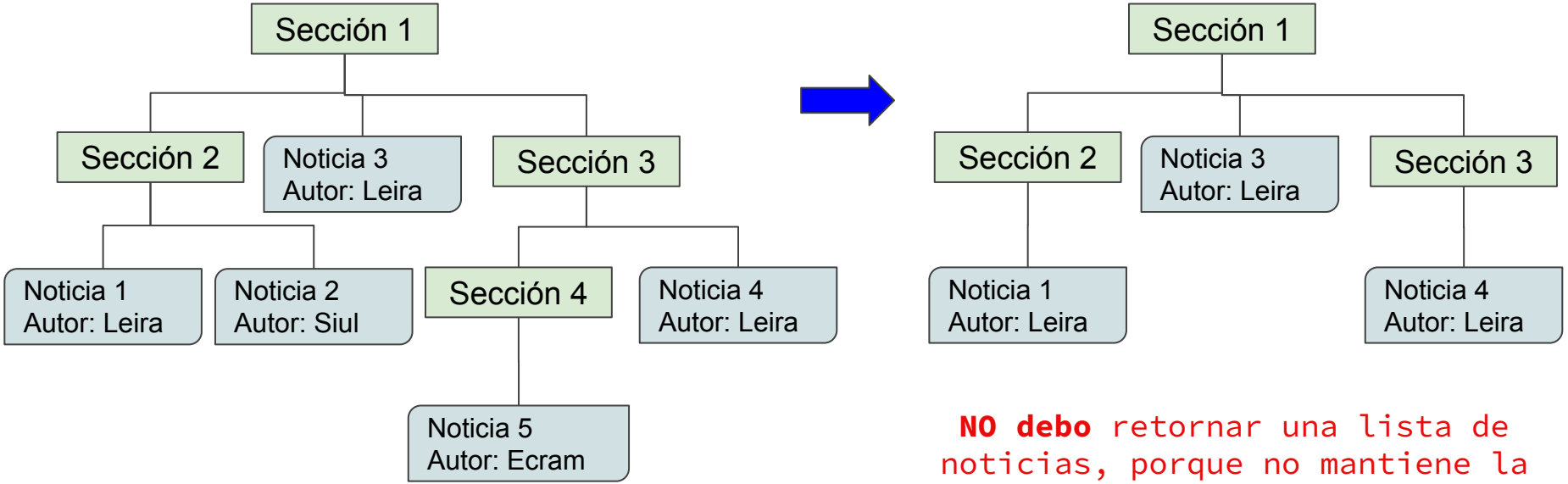
El portal ofrece un servicio de **restricción temática** para sus lectores. De esta forma, se debe poder obtener una copia de la estructura de noticias respetando la misma organización en secciones, subsecciones, grupos y subgrupos, pero restringida según diferentes criterios, por ejemplo:

- Que contenga solo las noticias cuyo título contenga la palabra “Tandil”
- Que contenga solo las noticias que en el contenido de la misma se encuentre la frase “en 1930”
- Que contenga solo las noticias de la categoría “Sociales” y cuyo autor NO sea “Ricardo Ruben”
- Que contenga solo las noticias que NO tengan la palabra clave “Qatar 2022”
- Cualquier combinación lógica de los anteriores

Tener en cuenta que, si una sección, subsección, grupo o subgrupo, no contiene ninguna noticia que cumpla con el criterio de restricción, la/el misma/o no se debe incluir en la copia. Considerar también que tanto las noticias como las secciones, subsecciones, grupos y subgrupos retornados por este servicio deben ser una copia del original, de tal forma que, si se modifican, el original se mantenga intacto.

Punto b)

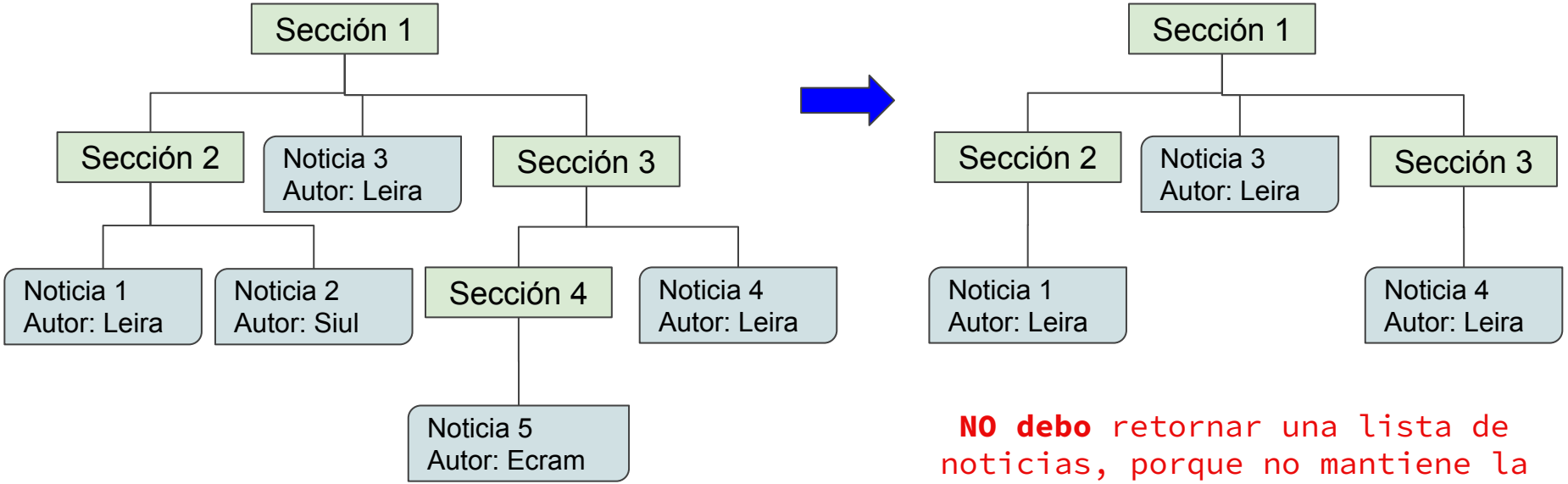
Si quiero las noticias cuyo autor sea Leira



NO debo retornar una lista de noticias, porque no mantiene la estructura

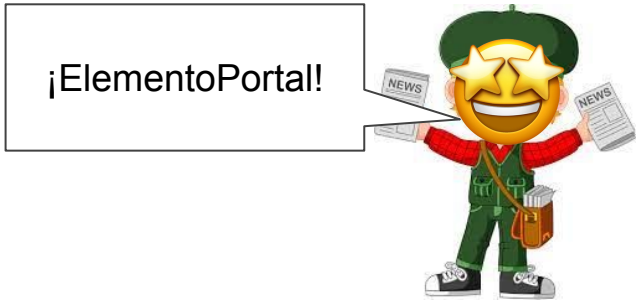
Punto b)

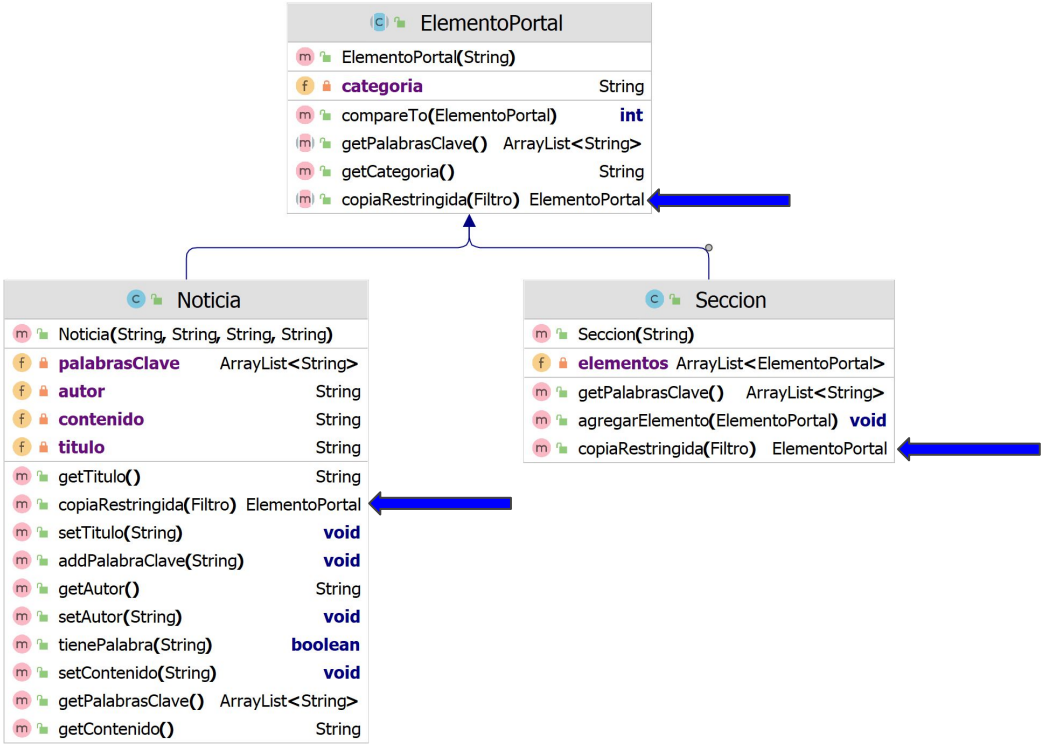
Si quiero las noticias cuyo autor sea Leira



NO debo retornar una lista de noticias, porque no mantiene la estructura

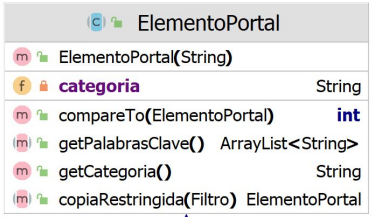
Punto b)

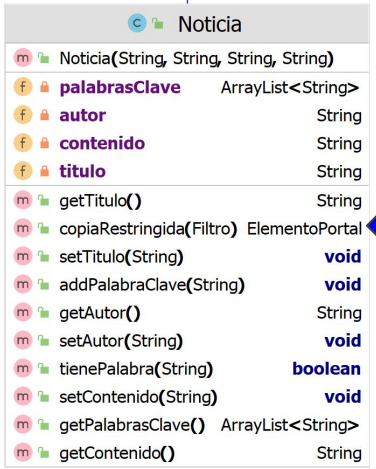

¡ElementoPortal!
NEWS
NEWS


ElementoPortal
ElementoPortal(String)
categoria String
compareTo(ElementoPortal) int
getPalabrasClave() ArrayList<String>
getCategoria() String
copiaRestringida(Filtro) ElementoPortal
Noticia
Noticia(String, String, String, String)
palabrasClave ArrayList<String>
autor String
contenido String
titulo String
getTitulo() String
copiaRestringida(Filtro) ElementoPortal
setTitulo(String) void
addPalabraClave(String) void
getAutor() String
setAutor(String) void
tienePalabra(String) boolean
setContenido(String) void
getPalabrasClave() ArrayList<String>
getContenido() String
Seccion
Seccion(String)
elementos ArrayList<ElementoPortal>
getPalabrasClave() ArrayList<String>
agregarElemento(ElementoPortal) void
copiaRestringida(Filtro) ElementoPortal

Punto b)

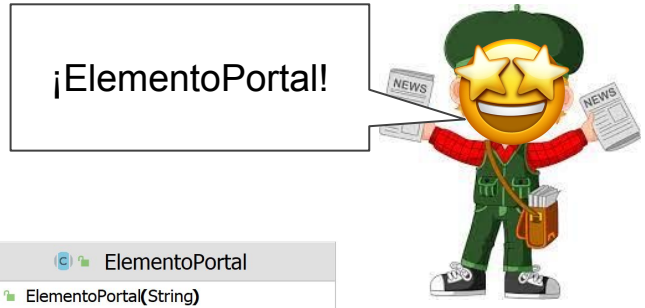
¡ElementoPortal!

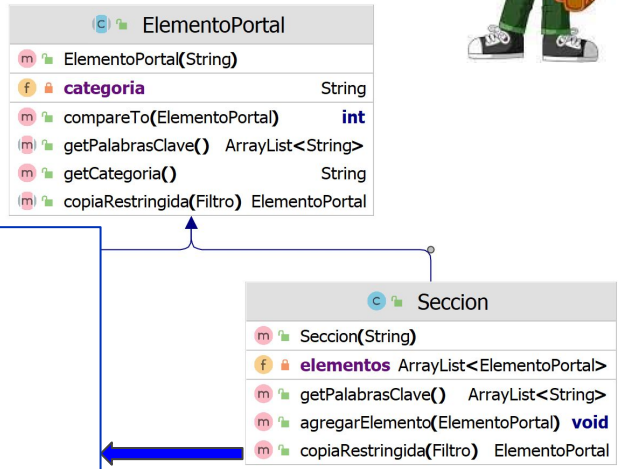





```java
@Override
public ElementoPortal copiaRestringida(Filtro filtro) {
    if (filtro.cumple( noticia: this)) {
        Noticia copia = new Noticia(this.titulo, this.contenido,
                this.autor, this.getCategoria());
        for (String s:this.palabrasClave)
            copia.addPalabraClave(s);
        return copia;
    }
    return null;
}
```

Punto b)





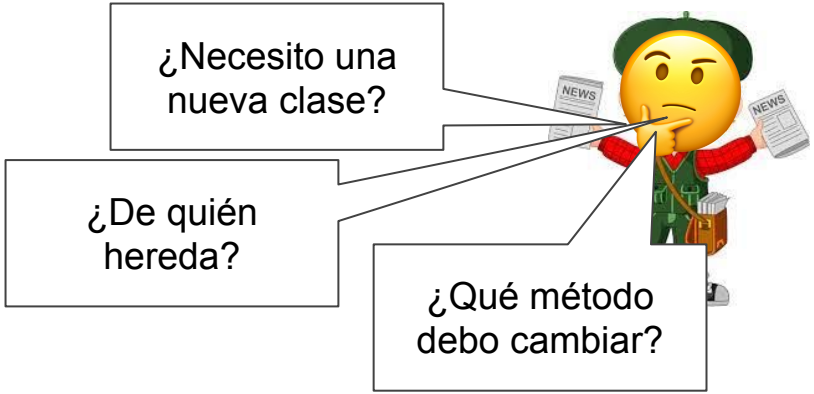

```java
@Override
public ElementoPortal copiaRestringida(Filtro filtro) {
    ArrayList<ElementoPortal> hijosQueCumplen = new ArrayList<>();
    for (ElementoPortal hijo:this.elementos){
        ElementoPortal copiaHijo = hijo.copiaRestringida(filtro);
        if (copiaHijo!=null)
            hijosQueCumplen.add(copiaHijo);
    }
    if (hijosQueCumplen.size()>0) { //algún hijo fue duplicado, debo retornar la sección
        Seccion copia = new Seccion(this.getCategoria());
        for (ElementoPortal copiaHijo:hijosQueCumplen)
            copia.agregarElemento(copiaHijo);
        return copia;
    }
    else //Ningún hijo fue duplicado, entonces retorno null (no duplico la sección)
        return null;
}
```

Punto c)

El dueño del portal desea agregar nuevos tipos de secciones y noticias:

- Una sección “*Policiales*” que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. Esta sección, solo permite que se agreguen elementos en ella que tenga una categoría que se encuentre en su listado de categorías admisibles <”robo”, “asesinato”, “accidente”>. Esta lista de categorías admisibles puede crecer a futuro, pero nunca eliminar categorías que antes se admitían.
- Una noticia ”Privada”, que tiene la particularidad de que nunca se incluye en una copia, cuando se solicita una restricción temática del portal.
- Un grupo “*Deportes*” que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. En esta sección deportes, solo se pueden agregar elementos cuya categoría sea una de las siguientes: <”Basquet”, ”futbol”, ”tenis”, ”bridge”>. El grupo “Deportes” puede aceptar a futuro elementos que posean otras categorías predefinidas, pero nunca restringir las que ya admitía.
- Una noticia “*Patrocinada*”, la cual posee un título, un contenido, palabras claves, el autor es “Leira Niresetnom”, y la categoría es “Negociación” (y no puede cambiar).

Punto c)



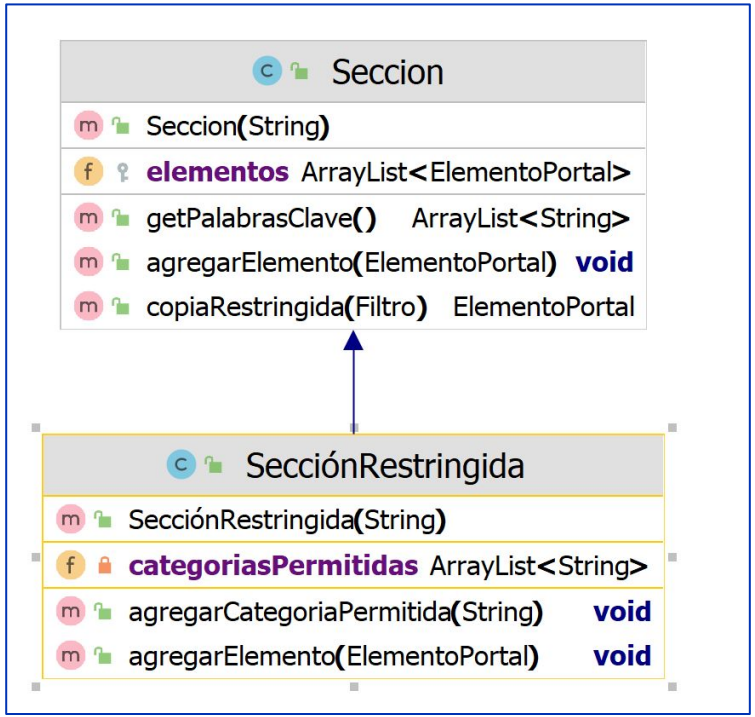
El dueño del portal desea agregar nuevos tipos de secciones y noticias:

- Una sección "*Policiales*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. Esta sección, **solo permite que se agreguen elementos en ella que tenga una categoría que se encuentre en su listado de categorías admisibles <"robo", "asesinato", "accidente">**. Esta lista de categorías admisibles puede crecer a futuro, pero nunca eliminar categorías que antes se admitían.
- Una noticia "Privada", que tiene la particularidad de que nunca se incluye en una copia, cuando se solicita una restricción temática del portal.
- Un grupo "*Deportes*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. En esta sección deportes, solo se pueden agregar elementos cuya categoría sea una de las siguientes: <"Basquet", "futbol", "tenis", "bridge">. El grupo "Deportes" puede aceptar a futuro elementos que posean otras categorías predefinidas, pero nunca restringir las que ya admitía.
- Una noticia "*Patrocinada*", la cual posee un título, un contenido, palabras claves, el autor es "Leira Niresetnom", y la categoría es "Negociación" (y no puede cambiar).

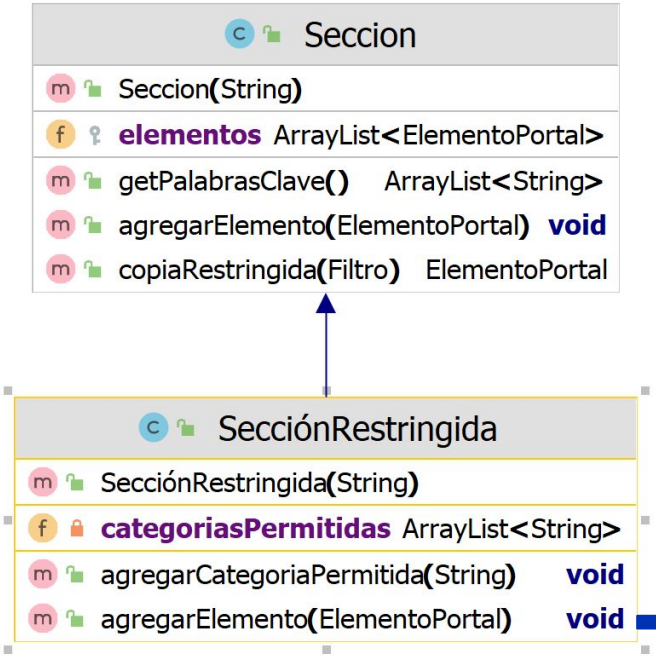
Punto c)

El dueño del portal desea agregar nuevos tipos de secciones y noticias:

- Una sección "*Policiales*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. Esta sección, **solo permite que se agreguen elementos en ella que tenga una categoría que se encuentre en su listado de categorías admisibles <"robo", "asesinato", "accidente">**. Esta lista de categorías admisibles puede crecer a futuro, pero nunca eliminar categorías que antes se admitían.
- Una noticia "Privada", que tiene la particularidad de que nunca se incluye en una copia, cuando se solicita una restricción temática del portal.
- Un grupo "*Deportes*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. En esta sección deportes, solo se pueden agregar elementos cuya categoría sea una de las siguientes: <"Basquet", "futbol", "tenis", "bridge">. El grupo "Deportes" puede aceptar a futuro elementos que



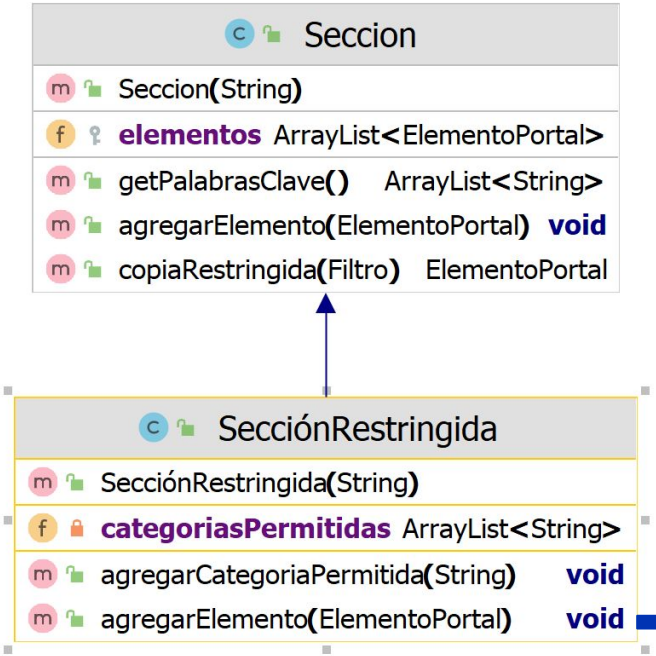
Punto c)




```java
@Override
public void agregarElemento(ElementoPortal nuevoElemento) {
    if (!categoriasPermitidas.contains(nuevoElemento.getCategoria())){
        int pos=0;
        while (pos<this.elementos.size() &&
                this.elementos.get(pos).compareTo(nuevoElemento)<0)
            pos++;
        this.elementos.add(pos, nuevoElemento);
    }
}
```

Punto c)

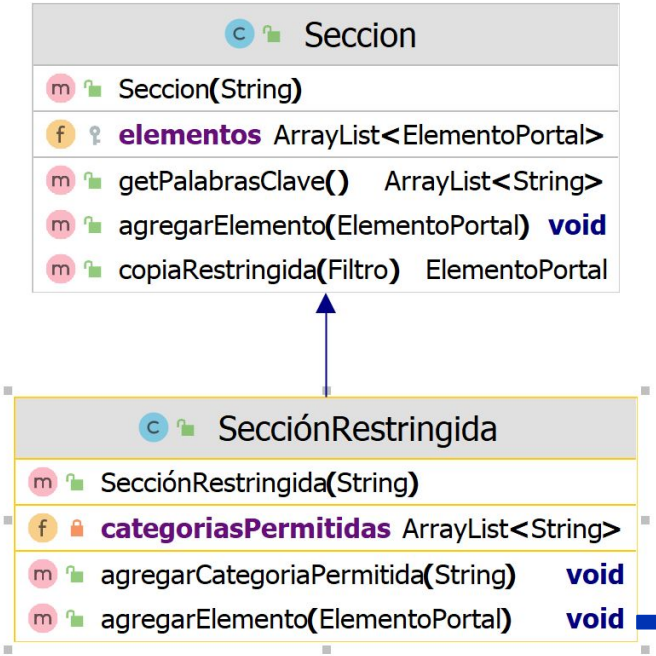
algo huele mal…



¡Todo esto lo hace la superclase!

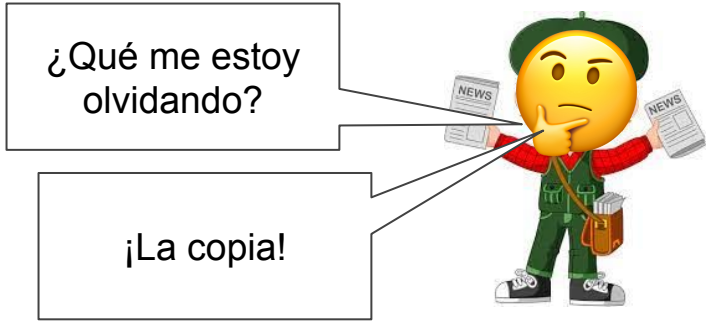
```
@Override
public void agregarElemento(ElementoPortal nuevoElemento) {
    if (!categoriasPermitidas.contains(nuevoElemento.getCategoria())){
        int pos=0;
        while (pos<this.elementos.size() &&
                this.elementos.get(pos).compareTo(nuevoElemento)<0)
            pos++;
        this.elementos.add(pos, nuevoElemento);
    }
}
```

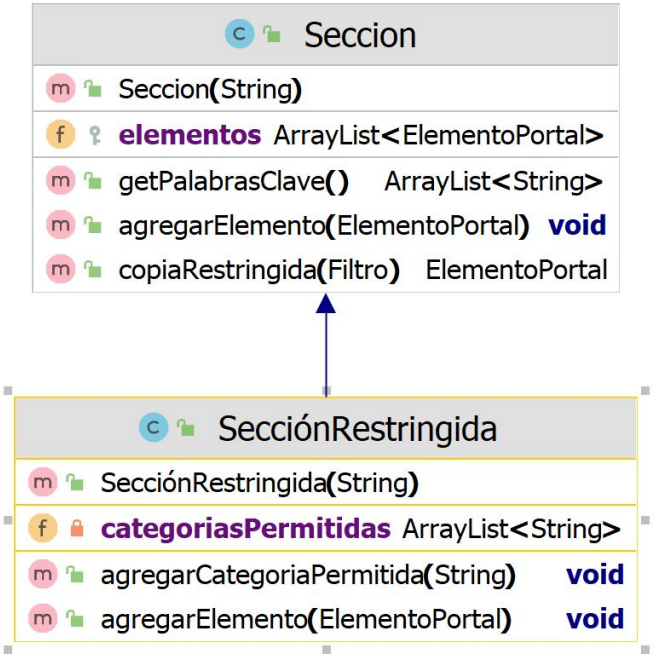
Punto c)




```java
@Override
public void agregarElemento(ElementoPortal nuevoElemento) {
    if (!categoriasPermitidas.contains(nuevoElemento.getCategoria()))
        super.agregarElemento(nuevoElemento);

}
```

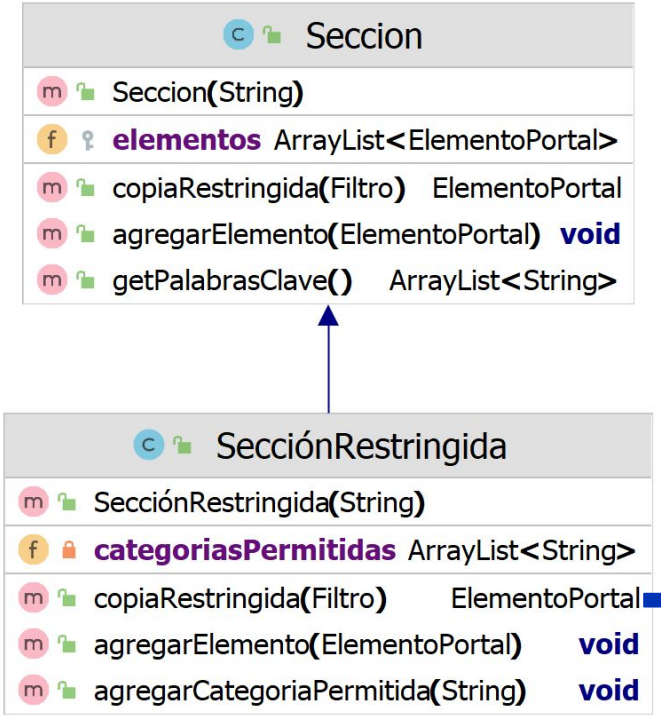
Punto c)





Si no redefino el método copiaRestringida, todas las secciones restringidas se convierten en secciones “comunes” al pedir una restricción temática

Punto c)




```java
@Override
public ElementoPortal copiaRestringida(Filtro filtro) {
    ArrayList<ElementoPortal> hijosQueCumplen = new ArrayList<>();
    for (ElementoPortal hijo:this.elementos){
        ElementoPortal copiaHijo = hijo.copiaRestringida(filtro);
        if (copiaHijo!=null)
            hijosQueCumplen.add(copiaHijo);
    }
    if (hijosQueCumplen.size()>0) { //algún hijo fue duplicado, debo retornar la sección
        SeccionRestringida copia = new SeccionRestringida(this.getCategoria());
        for (String categoria:copia.getCategoriasPermitidas())
            copia.agregarCategoriaPermitida(categoria);
        for (ElementoPortal copiaHijo:hijosQueCumplen)
            copia.agregarElemento(copiaHijo);
        return copia;
    }
    else //Ningún hijo fue duplicado, entonces retorno null (no duplico la sección)
        return null;
}
```

Punto c)

¿Se puede hacer mejor?

Seccion

- Seccion(String)
- **elementos** ArrayList<ElementoPortal>
- copiaRestringida(Filtro) ElementoPortal
- agregarElemento(ElementoPortal) **void**
- getPalabrasClave() ArrayList<String>
- obtenerCopia() Seccion

SeccionRestringida

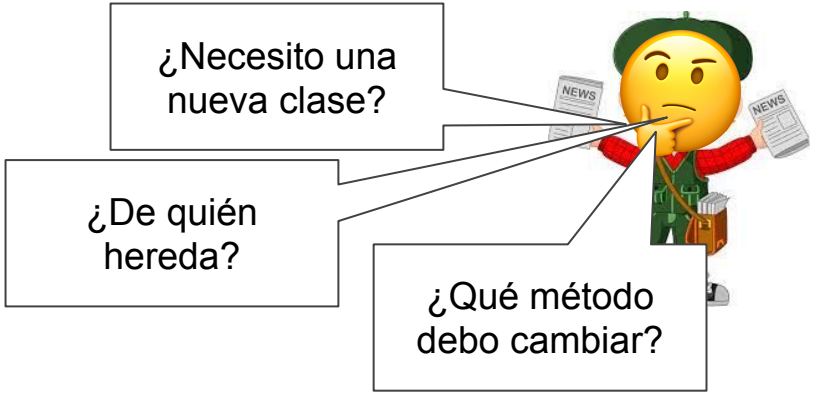
- SeccionRestringida(String)
- **categoriasPermitidas** ArrayList<String>
- obtenerCopia() Seccion
- agregarElemento(ElementoPortal) **void**
- agregarCategoriaPermitida(String) **void**

```java
@Override
public ElementoPortal copiaRestringida(Filtro filtro) {
    ArrayList<ElementoPortal> hijosQueCumplen = new ArrayList<>();
    for (ElementoPortal hijo:this.elementos){
        ElementoPortal copiaHijo = hijo.copiaRestringida(filtro);
        if (copiaHijo!=null)
            hijosQueCumplen.add(copiaHijo);
    }
    if (hijosQueCumplen.size()>0) { //algún hijo fue duplicado, debo retornar la sección
        Seccion copia = obtenerCopia();
        for (ElementoPortal copiaHijo:hijosQueCumplen)
            copia.agregarElemento(copiaHijo);
        return copia;
    }
    else //Ningún hijo fue duplicado, entonces retorno null (no duplico la sección)
        return null;
}
```

```java
public Seccion obtenerCopia() {
    return new Seccion(this.getCategoria());
}
```

```java
@Override
public Seccion obtenerCopia() {
    SeccionRestringida copia = new SeccionRestringida(this.getCategoria());
    for (String categoria:copia.getCategoriasPermitidas())
        copia.agregarCategoriaPermitida(categoria);
    return copia;
}
```

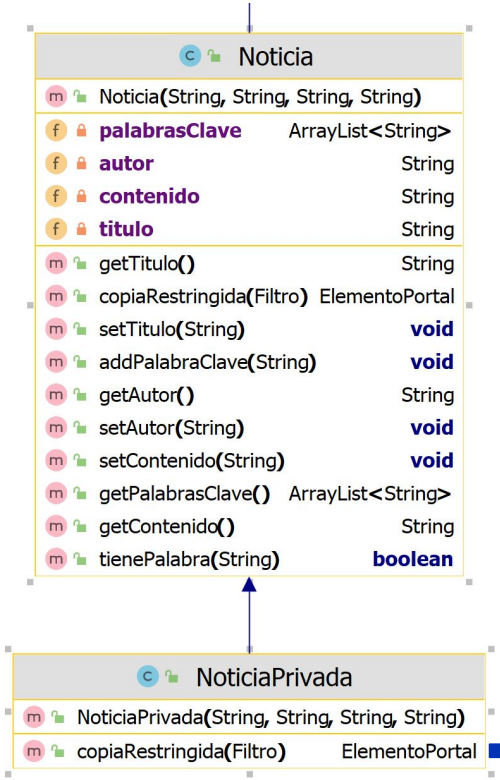
Punto c)



El dueño del portal desea agregar nuevos tipos de secciones y noticias:

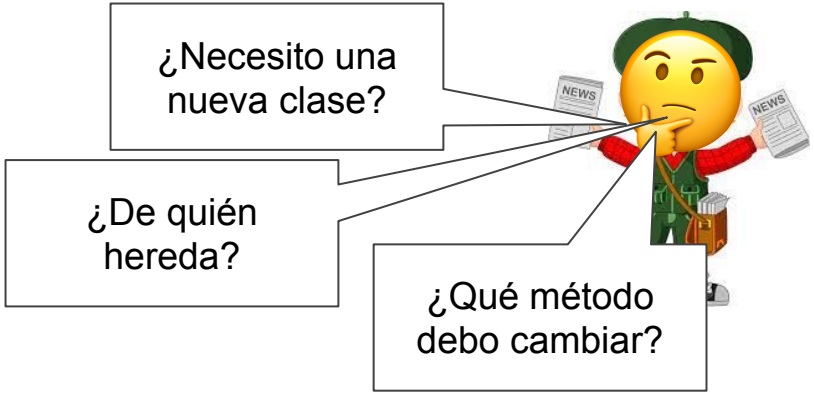
- Una sección "*Policiales*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. Esta sección, **solo permite que se agreguen elementos en ella que tenga una categoría que se encuentre en su listado de categorías admisibles <"robo", "asesinato", "accidente">**. Esta lista de categorías admisibles puede crecer a futuro, pero nunca eliminar categorías que antes se admitían.
- Una noticia "Privada", que tiene la particularidad de que **nunca se incluye en una copia**, cuando se solicita una restricción temática del portal.
- Un grupo "*Deportes*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. En esta sección deportes, solo se pueden agregar elementos cuya categoría sea una de las siguientes: <"Basquet", "futbol", "tenis", "bridge">. El grupo "Deportes" puede aceptar a futuro elementos que posean otras categorías predefinidas, pero nunca restringir las que ya admitía.
- Una noticia "*Patrocinada*", la cual posee un título, un contenido, palabras claves, el autor es "Leira Niresetnom", y la categoría es "Negociación" (y no puede cambiar).

Punto c)




```java
@Override
public ElementoPortal copiaRestringida(Filtro filtro) {
    return null;
}
```

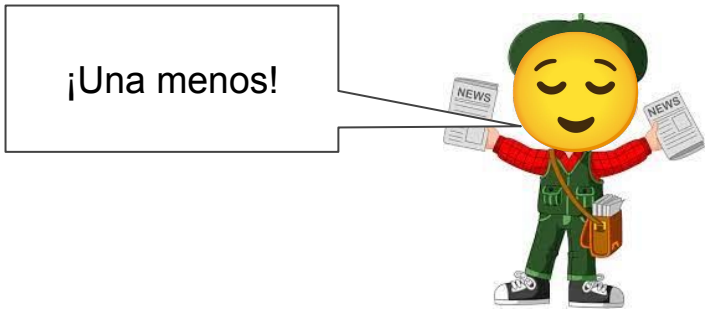
Punto c)



El dueño del portal desea agregar nuevos tipos de secciones y noticias:

- Una sección "*Policiales*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. Esta sección, **solo permite que se agreguen elementos en ella que tenga una categoría que se encuentre en su listado de categorías admisibles <"robo", "asesinato", "accidente">**. Esta lista de categorías admisibles puede crecer a futuro, pero nunca eliminar categorías que antes se admitían.
- Una noticia "Privada", que tiene la particularidad de que **nunca se incluye en una copia**, cuando se solicita una restricción temática del portal.
- Un **grupo** "*Deportes*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. En esta sección deportes, solo se pueden agregar elementos cuya categoría sea una de las siguientes: <"Basquet", "futbol", "tenis", "bridge">. El grupo "Deportes" puede aceptar a futuro elementos que posean otras categorías predefinidas, pero nunca restringir las que ya admitía.
- Una noticia "*Patrocinada*", la cual posee un título, un contenido, palabras claves, el autor es "Leira Niresetnom", y la categoría es "Negociación" (y no puede cambiar).

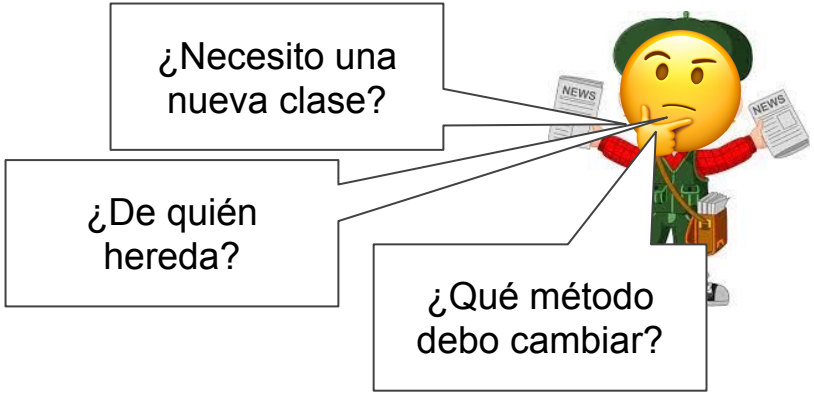
Punto c)



El dueño del portal desea agregar nuevos tipos de secciones y noticias:

- Una sección "*Policiales*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. Esta sección, **solo permite que se agreguen elementos en ella que tenga una categoría que se encuentre en su listado de categorías admisibles <"robo", "asesinato", "accidente">**. Esta lista de categorías admisibles puede crecer a futuro, pero nunca eliminar categorías que antes se admitían.
- Una noticia "Privada", que tiene la particularidad de que **nunca se incluye en una copia**, cuando se solicita una restricción temática del portal.
- ~~Un **grupo** "*Deportes*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. En esta sección deportes, solo se pueden agregar elementos cuya categoría sea una de las siguientes: <"Basquet", "futbol", "tenis", "bridge">. El grupo "Deportes" puede aceptar a futuro elementos que posean otras categorías predefinidas, pero nunca restringir las que ya admitía.~~
- Una noticia "*Patrocinada*", la cual posee un título, un contenido, palabras claves, el autor es "Leira Niresetnom", y la categoría es "Negociación" (y no puede cambiar).

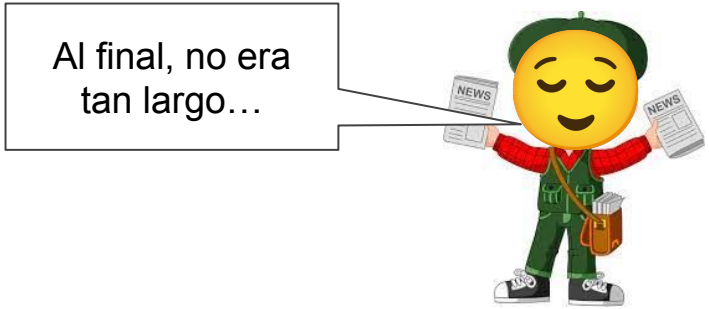
Punto c)



El dueño del portal desea agregar nuevos tipos de secciones y noticias:

- Una sección "*Policiales*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. Esta sección, **solo permite que se agreguen elementos en ella que tenga una categoría que se encuentre en su listado de categorías admisibles <"robo", "asesinato", "accidente"**>. Esta lista de categorías admisibles puede crecer a futuro, pero nunca eliminar categorías que antes se admitían.
- Una noticia "Privada", que tiene la particularidad de que **nunca se incluye en una copia**, cuando se solicita una restricción temática del portal.
- Un **grupo** "*Deportes*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. En esta sección deportes, solo se pueden agregar elementos cuya categoría sea una de las siguientes: <"Basquet", "futbol", "tenis", "bridge">. El grupo "Deportes" puede aceptar a futuro elementos que posean otras categorías predefinidas, pero nunca restringir las que ya admitía.
- Una noticia "*Patrocinada*", la cual posee un título, un contenido, palabras claves, el autor es "Leira Niresetnom", y la categoría es "Negociación" (y no puede cambiar).

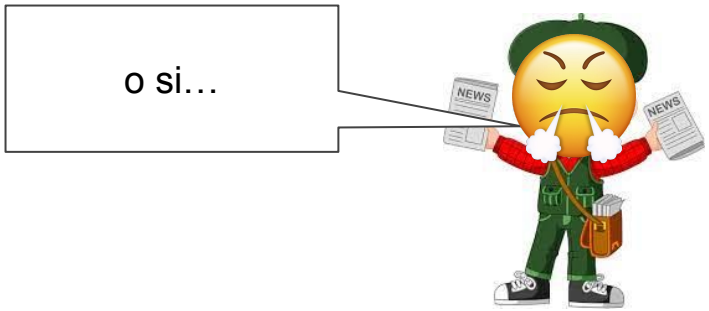
Punto c)



El dueño del portal desea agregar nuevos tipos de secciones y noticias:

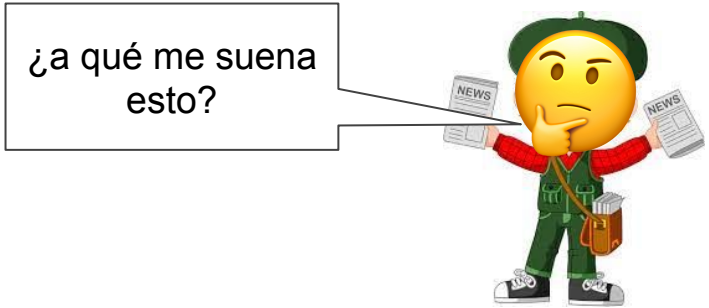
- Una sección "*Policiales*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. Esta sección, **solo permite que se agreguen elementos en ella que tenga una categoría que se encuentre en su listado de categorías admisibles <"robo", "asesinato", "accidente">**. Esta lista de categorías admisibles puede crecer a futuro, pero nunca eliminar categorías que antes se admitían.
- Una noticia "Privada", que tiene la particularidad de que **nunca se incluye en una copia**, cuando se solicita una restricción temática del portal.
- Un **grupo** "*Deportes*" que posee un conjunto de noticias, secciones, subsecciones, grupos y subgrupos. En esta sección deportes, solo se pueden agregar elementos cuya categoría sea una de las siguientes: <"Basquet", "futbol", "tenis", "bridge">. El grupo "Deportes" puede aceptar a futuro elementos que posean otras categorías predefinidas, pero nunca restringir las que ya admitía.
- ~~Una noticia "*Patrocinada*", la cual posee un título, un contenido, palabras claves, el autor es "Leira Niresetnom", y la categoría es "Negociación" (y no puede cambiar).~~

Punto d)



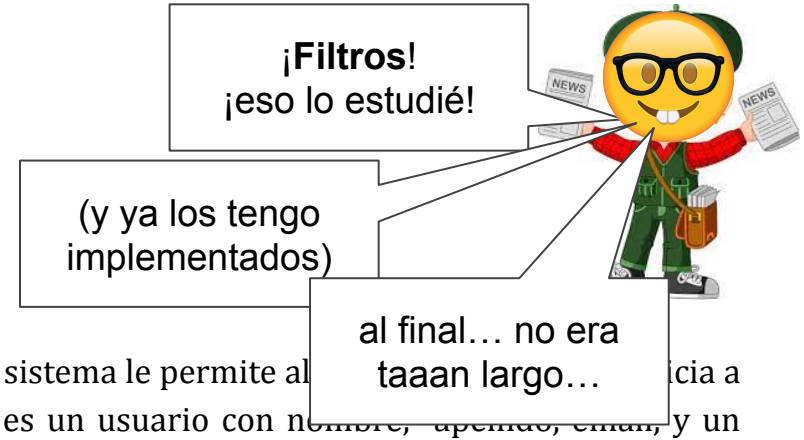
El portal desea incorporar un **sistema de suscripción a noticias**. Este sistema le permite al portal entregar una **noticia** a un listado de suscriptores interesados en la misma. Un **suscriptor** es un usuario con nombre, apellido, email, y un listado de noticias recibidas. A algunos suscriptores solo les interesan **noticias** que tengan la palabra “Miramar” en su título, mientras que a otros noticias que el autor sea “Leira Niresetnom”. Estos son solo algunos ejemplos de las políticas de suscripción a noticias de los usuarios y las mismas pueden combinarse e incluso cambiar en cualquier momento. Si a un suscriptor se le entrega una noticia que no le interesa, la misma NO es incorporada en su listado de noticias.

Punto d)



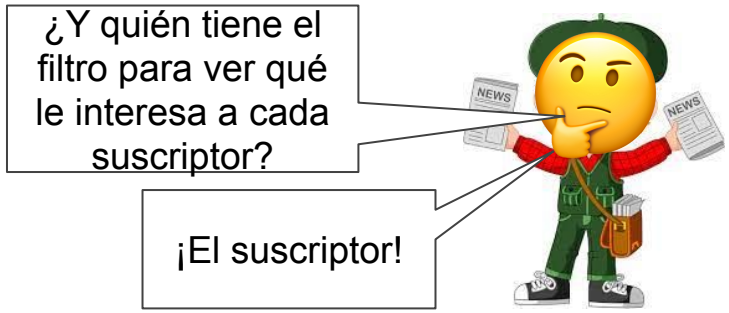
El portal desea incorporar un sistema de suscripción a noticias. Este sistema le permite al portal entregar una noticia a un listado de suscriptores interesados en la misma. Un suscriptor es un usuario con nombre, apellido, email, y un listado de noticias recibidas. **A algunos suscriptores solo les interesan noticias que tengan la palabra "Miramar" en su título, mientras que a otros noticias que el autor sea "Leira Niresetnom". Estos son solo algunos ejemplos de las políticas de suscripción a noticias de los usuarios y las mismas pueden combinarse e incluso cambiar en cualquier momento.** Si a un suscriptor se le entrega una noticia que no le interesa, la misma NO es incorporada en su listado de noticias.

Punto d)



El portal desea incorporar un sistema de suscripción a noticias. Este sistema le permite al [illegible]icia a un listado de suscriptores interesados en la misma. Un suscriptor es un usuario con nombre, apellido, email, y un listado de noticias recibidas. **A algunos suscriptores solo les interesan noticias que tengan la palabra “Miramar” en su título, mientras que a otros noticias que el autor sea “Leira Niresetnom”. Estos son solo algunos ejemplos de las políticas de suscripción a noticias de los usuarios y las mismas pueden combinarse e incluso cambiar en cualquier momento.** Si a un suscriptor se le entrega una noticia que no le interesa, la misma NO es incorporada en su listado de noticias.

Punto d)



El portal desea incorporar un sistema de suscripción a noticias. Este sistema le permite al portal entregar una noticia a un listado de suscriptores interesados en la misma. Un suscriptor es un usuario con nombre, apellido, email, y un listado de noticias recibidas. **A algunos suscriptores solo les interesan noticias que tengan la palabra "Miramar" en su título, mientras que a otros noticias que el autor sea "Leira Niresetnom". Estos son solo algunos ejemplos de las políticas de suscripción a noticias de los usuarios y las mismas pueden combinarse e incluso cambiar en cualquier momento.** Si a un suscriptor se le entrega una noticia que no le interesa, la misma NO es incorporada en su listado de noticias.

Punto d)

¡Strategy!

Suscriptor	
Suscriptor(String, String, String, Filtro)	
apellido	String
email	String
noticiasRecibidas	ArrayList<Noticia>
nombre	String
interes	Filtro
recibirNoticia(Noticia)	void
cambiarInteres(Filtro)	void
agregarInteres(Filtro)	void

El filtro representa los intereses del suscriptor (que pueden cambiar)

¿Cómo se usa?

```
Filtro contenidoMiramar = new FiltroContenido( substringBuscado: "Miramar");
Filtro autorLeiraNiresetnom = new FiltroAutor( autorBuscado: "Leira Niresetnom");
Filtro intereses = new FiltroOr(contenidoMiramar, autorLeiraNiresetnom);

Suscriptor lector = new Suscriptor( nombre: "Juan",  apellido: "Perez",  email: "jperez@gmail.com", intereses);
```

Punto d)

Suscriptor	
Suscriptor(String, String, String, Filtro)	
apellido	String
email	String
noticiasRecibidas	ArrayList<Noticia>
nombre	String
interes	Filtro
recibirNoticia(Noticia)	void
cambiarInteres(Filtro)	void
agregarInteres(Filtro)	void

```
public void cambiarInteres(Filtro nuevoInteres){
    this.interes = nuevoInteres;
}
```

Punto d)

Suscriptor	
Suscriptor(String, String, String, Filtro)	
apellido	String
email	String
noticiasRecibidas	ArrayList<Noticia>
nombre	String
interes	Filtro
recibirNoticia(Noticia)	void
cambiarInteres(Filtro)	void
agregarInteres(Filtro)	void

(opcional, no se pedía)

```
public void agregarInteres(Filtro nuevoInteres){
    Filtro and = new FiltroAnd(this.interes, nuevoInteres);
    this.cambiarInteres(and);
}
```

Punto d)

Este sí es el que se pedía

Suscriptor	
Suscriptor(String, String, String, Filtro)	
apellido	String
email	String
noticiasRecibidas	ArrayList<Noticia>
nombre	String
interes	Filtro
recibirNoticia(Noticia)	void
cambiarInteres(Filtro)	void
agregarInteres(Filtro)	void

```
public void recibirNoticia(Noticia noticia){
    if (interes.cumple(noticia))
        this.noticiasRecibidas.add(noticia);
}
```

Punto d)

Este sí es el que se pedía

Portal	
Portal(ElementoPortal)	
categoriaRaiz	ElementoPortal
suscriptores	ArrayList<Suscriptor>
repartirNoticiaASuscriptores(Noticia)	void

```
public void repartirNoticiaASuscriptores(Noticia noticia){
    for (Suscriptor suscriptor:suscriptores)
        suscriptor.recibirNoticia(noticia);
}
```

Solución final