

# Metodologías de Programación I

---

## Héroes de Ciudad

### Patrón Iterator

#### Propósito

Implementar la atención de denuncias de incendios en el cuartel de bomberos.

#### Denuncias

En el cuartel de bomberos una persona atiende las denuncias de las personas que observan un principio de incendio, tanto de una casa como de una plaza. Esta persona está atenta a varios mecanismos

- Un tablero que enciende sus luces cuando una alarma contra incendios comienza a sonar.
- Una PC donde las denuncias llegan por whatsapp.
- Un mostrador para atender a las personas que se acercan al cuartel de bomberos.

#### Implementación

- 1) Implemente una interface *IResponsable* (por ahora sin ningún método).
- 2) Haga que *Bombero* implemente la interface *IResponsable*.
- 3) Implemente una interface *IDenuncia* con el método:
  - a. `void atender(IResponsable unResponsable)`
- 4) Implemente una clase *DenunciaDeIncendio* con la interface *IDenuncia*. Esta clase debería almacenar en su estado el *ILugar* que se está incendiando para que el responsable (un *Bombero*) apague el fuego cuando se invoque al método *atender* de esta clase.
- 5) Implemente una interface *IDenuncias* que represente a un conjunto de denuncias (*IDenuncia*) de aparición de incendios. Implemente tres clases con esta interface: *DenunciasPorTablero*, *DenunciasPorWhatsapp*, *DenunciasPorMostrador*.
- 6) La clase *DenunciasPorTablero* tiene como parte de su estado una *List<IDenuncia>* con todos los *ILugar* donde se activó la alarma de incendios. Para ello haga que esta clase sea observadora de los *ILugar* que tienen instalado una alarma contra incendios y cuando esta clase recibe la notificación (método *actualizar* del observador) agrega a la lista una nueva *DenunciaDeIncendio* con el *ILugar* donde sonó la alarma.
- 7) La clase *DenunciasPorWhatsapp* tiene una lista enlazada de los *ILugar* donde los habitantes observan el comienzo de un incendio. Para crear una lista enlazada puede usar una clase auxiliar (*MensajeWhatsapp*) que tenga como estado una *IDenuncia* con el *ILugar* afectado y el siguiente mensaje de la lista. Para armarla puede usar el mismo mecanismo usado en Decorator para armar los decorados:

```
MensajeWhatsapp lista = null;
lista = new MensajeWhatsapp(new DenunciaDeIncendio(lugar1), lista);
lista = new MensajeWhatsapp(new DenunciaDeIncendio(lugar2), lista);
lista = new MensajeWhatsapp(new DenunciaDeIncendio(lugar3), lista);
lista = new MensajeWhatsapp(new DenunciaDeIncendio(lugar4), lista);
DenunciasPorWhatsapp denuncia = new DenunciasPorWhatsapp(lista);
```

- 8) La clase *DenunciasPorMostrador* tiene una única variable *IDenuncia* con el *ILugar* donde el denunciante observó el inicio de un incendio.
- 9) Implemente con Iterator un iterador de *IDenuncia* para cada clase:  
*DenunciasPorTablero*, *DenunciasPorWhatsapp*, *DenunciasPorMostrador*
- 10) Implemente una clase *BomberoSecretario* que tenga como estado un *Bombero* (quien será el único que apague todos los incendios) y que implemente el método:
  - a. void atenderDenuncias(IDenuncias denuncias)

Este método deberá pedir el iterador a la *IDenuncias* recibida por parámetro y por cada *IDenuncia* iterada invocar a su método atender con el bombero almacenado en el estado.

## Testing

Implemente una función *main* en la clase *HeroesDeCiudad* para que instancie un *BomberoSecretario*, un *Bombero* (con cualquier estrategia) y diez *ILugar* cualquiera (casa o plaza), llamémoslos: A, B, C, D, E, F, G, H, I y J.

Instancie una *DenunciasPorTablero* y hágala observadora de los lugares A, B, C, D, E y F.

Instancie una *DenunciasPorWhatsapp* y agréguele los lugares G, H e I.

Instancie una *DenunciasPorMostrador* con el lugar J.

Invoque al método *chispa* de los lugares B y F (para que se agreguen denuncias a la lista de *DenunciasPorTablero*).

Invoque al método *atenderDenuncias* del *BomberoSecretario* con las tres *IDenuncias* instanciadas.

Compruebe el correcto funcionamiento del sistema.