

# ENS IV – Tasca 03 part 1

Autor: Sergio Blanco Cuaresma

## Enunciat

Modifica el ProjectileApplet per afegir els següents requeriments:

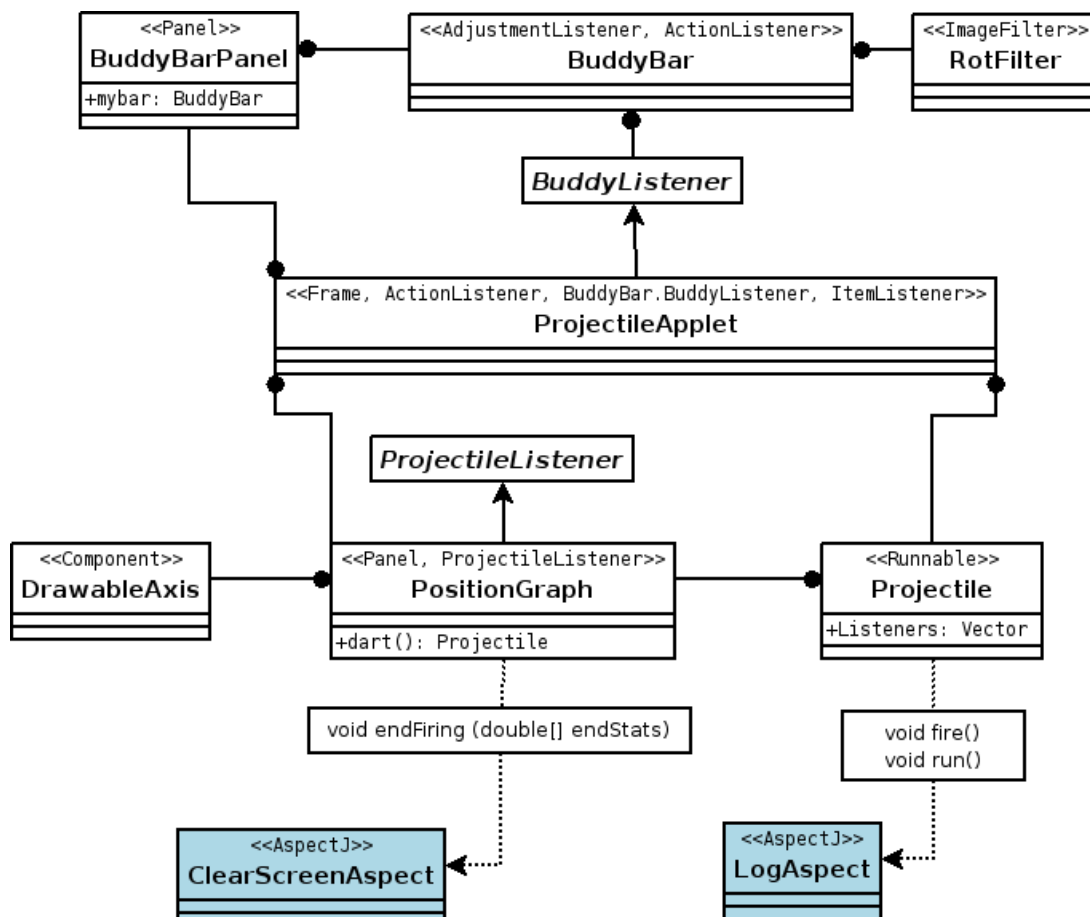
- Afegeix un Logger que enregistri per cada llançament els valors inicials (velocitat, angle, massa, resistència a l'aire) i els valors finals (distància màxima, alçada màxima, velocitat final, temps final).
- Quan finalitza un llançament ha de preguntar a l'usuari si vol netejar l'històric de llançaments.

## Notes

La pràctica ha sigut desenvolupada utilitzant Eclipse amb el plugin AJDT (<http://eclipse.org/ajdt/>) per fer ús d'AspectJ (<http://eclipse.org/aspectj/>).

## Disseny

Per complir els requeriments, he implementat dos Aspects: LogAspect i ClearScreenAspect.



## Implementació

### LogAspect

L'aspecte LogAspect s'encarrega de guardar en un fitxer les dades inicials (velocitat, angle, massa, resistència a l'aire) i finals (distància màxima, alçada màxima, velocitat final, temps finals) dels tirs.

Per això estableix dos pointcuts (per evitar duplicitats als logs, cal no ficar el punts a crides d'interfície):

- Abans de l'execució del mètode “void fire()” de Projectile.
- Abans de l'execució del mètode “void run()” de Projectile.

A continuació tenim els advices:

```
//// Inici tir
pointcut firePointCut(Projectile p) :
    execution(void Projectile.fire()) && this(p);

before(Projectile p): firePointCut(p)
{
    double velocitat = p.velocityInit;
    double angle = p.angle;
    double massa = p.m;
    double resistencia = p.k;
    escriureIniciLog(velocitat, angle, massa, resistencia);
}

//// Final tir
pointcut runPointCut(Projectile p) :
    execution(void Projectile.run()) && this(p);

after(Projectile p): runPointCut(p)
{
    if (p.hit) {
        double distancia = p.x*p.x0;
        double velocitat = Math.sqrt
            (p.vx*p.vx+p.vy*p.vy);
        double alcadaMax = p.maxHeight*p.y0;
        double tempsFinal = p.t*p.t0;
        escriureFinalLog(distancia, velocitat, alcadaMax, tempsFinal);
    }
}
```

Resultat del log al fitxer “Log.txt”:

```
-----
Inici Tir #1
- Velocitat = 50.0
- Angle = 1.2217304763960306
- Massa = 10.0
- Resistencia = 0.0

Final Tir #1
- Distancia final = 163.80927871726297
```

- Velocitat final = 50.000000000000006
- Alçada maxima = 112.50837259649708
- Temps final = 9.578925798021498

-----  
Inici Tir #2

- Velocitat = 50.0
- Angle = 1.2217304763960306
- Massa = 19.05
- Resistencia = 0.0

Final Tir #2

- Distancia final = 163.80927871726297
- Velocitat final = 50.000000000000006
- Alçada maxima = 112.50837259649708
- Temps final = 9.578925798021498

## ClearScreenAspect

L'aspecte ClearScreenAspect s'encarrega de preguntar a l'usuari si vol que sigui netejada la pantalla després de realitzar el tir. Per això s'estableix un únic pointcut:

- Després de l'execució del mètode “void endFiring(double[] endStats)” de PositionGraph.

A continuació tenim l'advice:

```
pointcut clearScreenPointCut(PositionGraph p):  
    execution(void PositionGraph.endFiring(double[])) && this(p);  
  
after (PositionGraph p): clearScreenPointCut(p)  
{  
    int resposta = JOptionPane.showConfirmDialog(p, "Vols netejar la pantalla  
de llençaments?", "Neteja de pantalla", JOptionPane.YES_NO_CANCEL_OPTION);  
    if (resposta == JOptionPane.YES_OPTION)  
    {  
        p.shotList.clear();  
        p.repaint();  
    }  
}
```

Resultat:

