

CICLO FINAL

JULIAN CAMILO LOPEZ BARRERO

JUAN SEBASTIAN PUENTES JULIO

ESCUELA COLOMBIANA DE INGENIERÍA JULIO GARAVITO

POOB

BOGOTÁ, COLOMBIA

05 DE ABRIL DE 2025

ESCUELA COLOMBIANA DE INGENIERÍA

PROGRAMACIÓN ORIENTADA A OBJETOS

PROYECTO INICIAL Cierre 2025-1

REFACTORING

El proyecto inicial tiene como propósito desarrollar una aplicación que permita simular una situación inspirada en el Problem H de la maratón de programación internacional 2024 Maxwell's Demon. En este simulador se desean tener varios demonios, identificar las partículas por su color (máximo 50 colores) y adicionar agujeros negros por los que pueden desaparecer un número limitado de partículas.

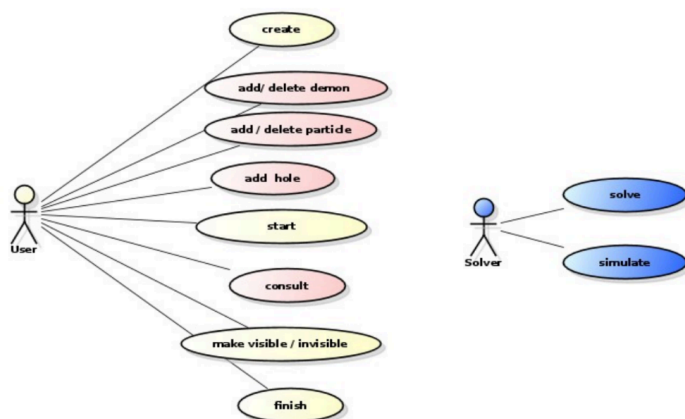
CIERRE

El objetivo de esta entrega es perfeccionar el simulador cumpliendo los criterios de calidad de funcionalidad y extensibilidad usando las facilidades que ofrece el IDE eclipse.

Para esto se espera que en esta entrega se tenga:

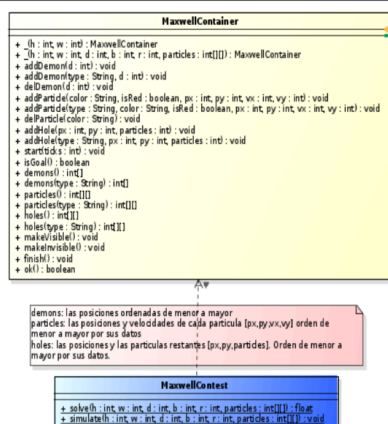
1. El proyecto en IDE eclipse
2. Un informe de análisis dinámico: informe de cubrimiento de pruebas de Junit.
3. Un buen informe de análisis estático: informe de PMD. Reglas java.

REQUISITOS FUNCIONALES



REQUISITOS DE DISEÑO Y CONSTRUCCIÓN

REQUISITOS DE DISEÑO Y CONSTRUCCIÓN



REQUISITOS DE ENTREGABLES

Los productos esperados para esta entrega son:

1. Diseño completo en la herramienta astah.

Adicionar el diagrama de paquetes.

En astah, crear un diagrama de clases (cambiar el nombre por Package Diagram0)

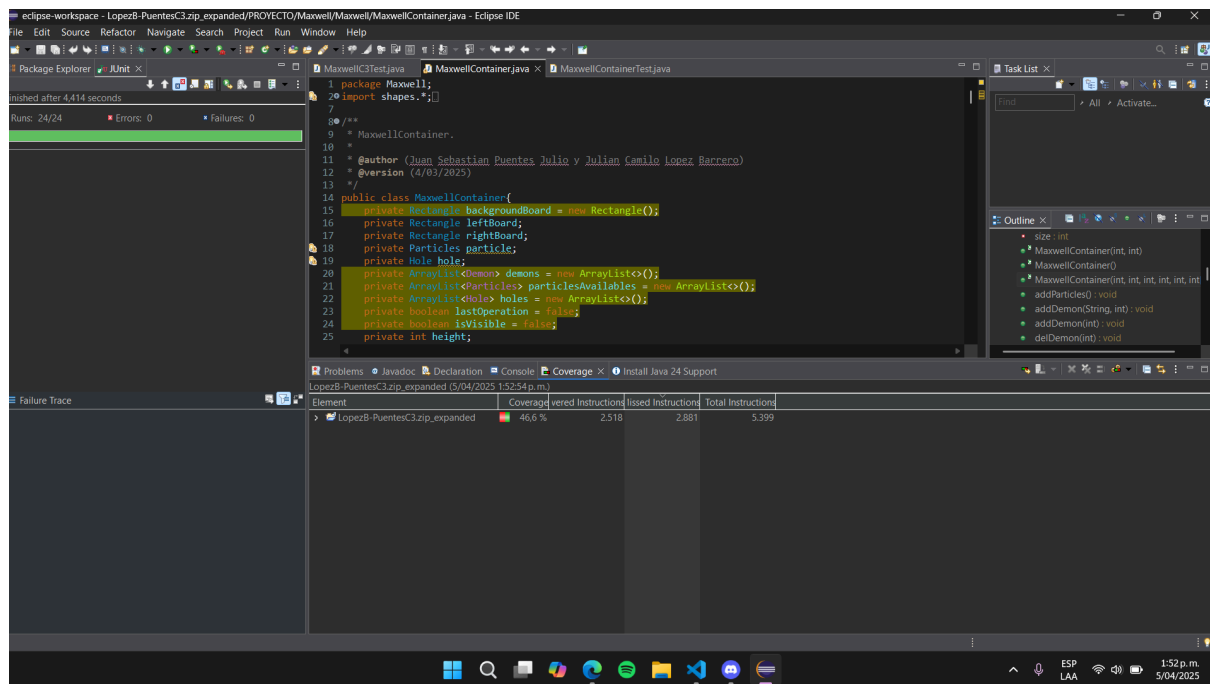
2. Código completo en el IDE eclipse.

3. Código de pruebas de unidad completas en el IDE eclipse.

4. Código de pruebas de aceptación en el IDE eclipse.

5. Informes de análisis dinámico; resultado inicial y resultado final.

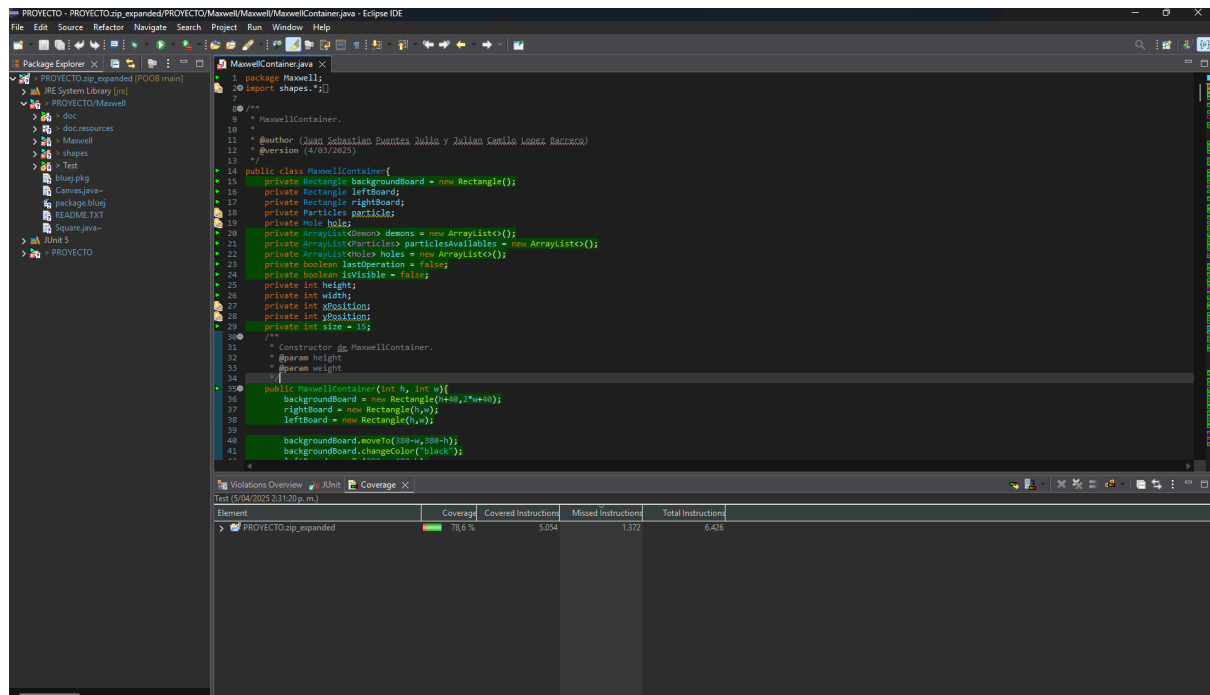
Resultado Inicial



Descripción de las decisiones tomadas después de estudiar el resultado inicial.

Tras ver el coverage inicial lo que decidimos usar fue realizar pruebas donde estas involucren una abundante cantidad de código para así tener el porcentaje que quería

Resultado Final



Meta: más de 75% de cubrimiento de código de dominio.

6. Infomes de análisis estático: resultado inicial y resultado final.

Tras el análisis estático observamos que teníamos 3 errores graves los cuales eran causados por la firma de 3 métodos.

Maxwell	3	2.9	0.03	PROYECTO.zip_...
MaxwellContainer.java	3	6.3	0.07	PROYECTO.zip_...

```

76 public MaxwellContainer(int h, int w, int d, int b, int r, int[][] particles) {
77     if(h<0) lastOperation = false;
78     backgroundBoard = new Rectangle(h+40,2*w+40);
79     rightBoard = new Rectangle(h,w);
80     leftBoard = new Rectangle(h,w);
81
82     backgroundBoard.moveTo(380-w,380-h);
83     backgroundBoard.changeColor("black");
84     leftBoard.moveTo(390-w,400-h);
85     rightBoard.moveTo(410,400-h);
86
87     width = w;
88     height = h;
89     lastOperation = true;
90     addDemon(d);
91     int total = b+r;
92     for (int i = 0; i < particles.length; i++) {
93         if (total <= r) {
94             addParticle("blue", false, particles[i][0], particles[i][1], particles[i][2], particles[i][3]);
95             continue;
96         }
97         addParticle("red", true, particles[i][0], particles[i][1], particles[i][2], particles[i][3]);
98         total--;
99     }
100 }

```

Descripción de las decisiones tomadas después de estudiar el resultado inicial.

Tras ver estos errores a los métodos addDemon, AddParticle y conversionBoardToCanvas decidimos ponerle la firma a “final” y así los errores fueron solucionados.

```

public final void addDemon(int d) {
    if (0 <= d && d <= height - 20) {
        for (Demon existingDemon : demons) {
            int y = conversionBoardToCanvas(0,d).get(1) - 20 ;
            if (Math.abs(existingDemon.getY() - y) < 20) {
                JOptionPane.showMessageDialog(null, "There is already a Demon in that position");
                lastOperation = false;
                return;
            }
        }
        int y = conversionBoardToCanvas(0,d).get(1) - 20 ;
        int x = conversionBoardToCanvas(0,d).get(0);
        Demon demon = new Demon(x,y);
        demon.moveDemonTo(x,y);
        demons.add(demon);
        lastOperation = true;
    }
    else {
        JOptionPane.showMessageDialog(null, "Invalid Range");
        lastOperation = false;
    }
}
}

```

```

public final void addParticle(String color,boolean isRed, int px, int py, int vx, int vy){
    if (isInLeft(px,py)){
        createInLeft("Normal",color,isRed, px,py,vx,vy);
        lastOperation = true;
    }
    else if (isInRight(px,py)){
        createInRight("Normal",color, isRed, px,py,vx,vy);
        lastOperation = true;
    }
}
}

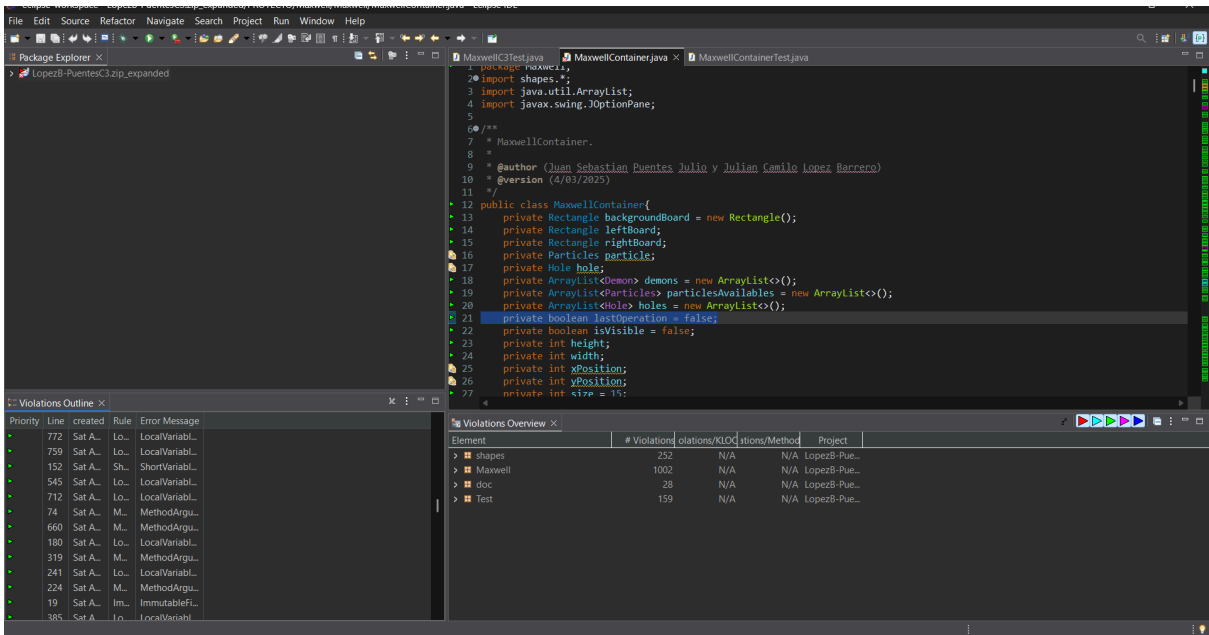
```

```

public final ArrayList<Integer> conversionBoardToCanvas(int x, int y){
    ArrayList<Integer> conversions = new ArrayList<>();
    int chamberXPos = leftBoard.getX() + width;
    int chamberYPos = leftBoard.getY() + height;
    int positionXEsperada = x + chamberXPos;
    int positionYESperada = chamberYPos - y;
    conversions.add(positionXEsperada);
    conversions.add(positionYESperada);
    return conversions;
}
}

```

Meta: cumplimiento de todas las reglas de prioridad alta (rojo).



Priority	Line	created	Rule	Error Message
▶	524	Sat A...	Sy...	SystemPrintl...
▶	534	Sat A...	Sy...	SystemPrintl...
▶	152	Sat A...	M...	MethodArgu...
▶	670	Sat A...	Lit...	LiteralsFirstl...
▶	387	Sat A...	Lit...	LiteralsFirstl...
▶	224	Sat A...	Sh...	ShortVariabl...
▶	258	Sat A...	Lo...	LocalVariabl...
▶	24	Sat A...	Im...	ImmutableFi...
▶	224	Sat A...	M...	MethodArgu...
▶	734	Sat A...	Lit...	LiteralsFirstl...
▶	203	Sat A...	Sh...	ShortVariabl...
▶	639	Sat A...	M...	MethodArgu...
▶	711	Sat A...	Lo...	LocalVariabl...

7. Documento de retrospectiva. (8 preguntas ver ciclo uno)

Es necesario incluir la retrospectiva de este ciclo y de los anteriores.

REQUISITOS DE ENTREGA

Los productos los deben publicar en el espacio preparado en moodle en un archivo .zip con un nombre igual a la concatenación de los apellidos de los autores, ordenados alfabéticamente.