

MANUAL DE USUARIO

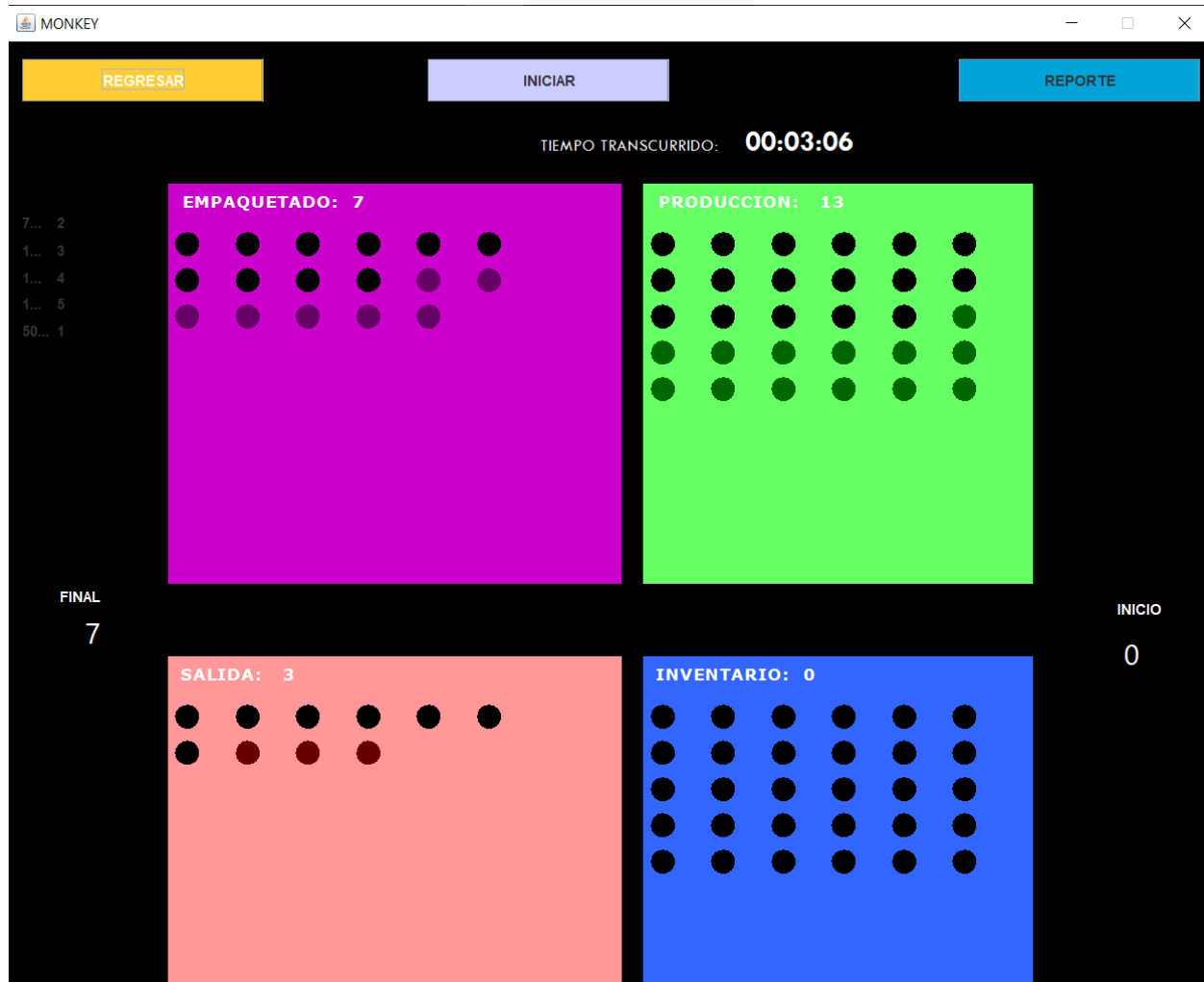
MENU MONKEY

BIENVENIDO A MONKEY

	TIEMPO (s)	COSTO (Q/s)
INVENTARIO	<input type="text"/> seg	Q. <input type="text"/>
PRODUCCION:	<input type="text"/> seg	Q. <input type="text"/>
EMPAQUETADO:	<input type="text"/> seg	Q. <input type="text"/>
SALIDA:	<input type="text"/> seg	Q. <input type="text"/>

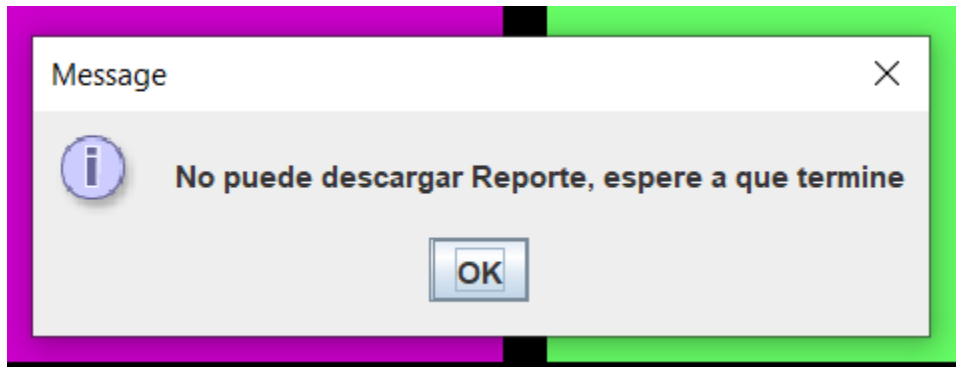
INICIAR SIMULACION

Se muestra un panel en el cual se deben llenar con datos de números enteros, ya que no es muy conveniente trabajar con tiempos decimales o por otro lado tiempos negativos, y en la siguiente columna se deben ingresar los datos del precio que servirá posteriormente en el reporte

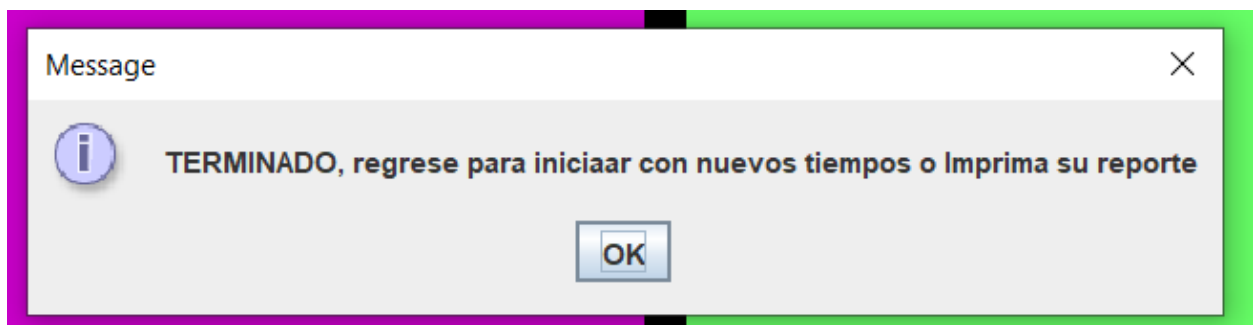


Según los tiempos ingresados se tardaran ese tiempo en esa región para que se vea lo que esta pasando se muestran puntos del color del panel que les corresponde y para no perder dirección de lo que se hace se pintaron de color negro

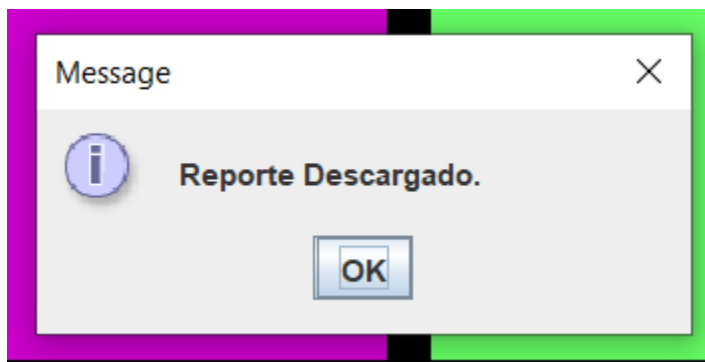
SEÑALES DE ALERTA



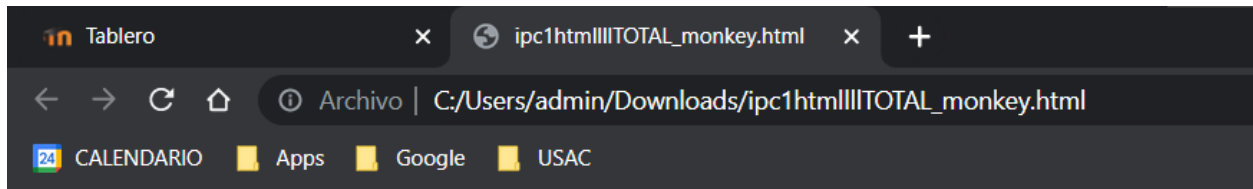
Si la simulación ha comenzado no hay nada que la detenga, solo hasta que finalice, de lo contrario mostrara esta alerta



Bien al terminar muestra esta alerta diciendo que ha terminado y pausara los hilos de los contadores



Es señal de confirmación cuando se presiona el reporte



Reporte Costo De Simulación

Costo Inventario	Q. 720.0.00
Costo Producción	Q. 1080.0.00
Costo Empaquetado	Q. 1440.0.00
Costo Salida	Q. 1800.0.00
Costo Total	Q. 5040.0.00

PABLO ANDRES RODRIGUEZ LIMA - 202201947

El reporte muestra una tabla de cual es el precio por entrar a cada simulación, también un total de la simulación x/segundo

MANUAL TECNICO

BIENVENIDO A MONKEY

	TIEMPO (s)	COSTO (Q/s)
INVENTARIO	<input type="text"/> seg	Q. <input type="text"/>
PRODUCCION:	<input type="text"/> seg	Q. <input type="text"/>
EMPAQUETADO:	<input type="text"/> seg	Q. <input type="text"/>
SALIDA:	<input type="text"/> seg	Q. <input type="text"/>

INICIAR SIMULACION

En el form del menú se validan algunas situaciones tanto que si no es un entero positivo te da la opción a repetir, esto hecho con un condicional if, y también se hacen las conversiones para mostrarlos en el reporte y también en la utilización de los tiempos en los paneles.

```

if(inventariotiempoTF.getText().isEmpty() || producciontiempoTF.getText().isEmpty() || empaquetadotiempoTF.getText().isEmpty() || salidatiempoTF.getText().isEmpty())
{
    JOptionPane.showMessageDialog(null, "CAMPOS VACIOS, ASEGURE DE LLENAR TODOS, VERIFIQUE Y AGREGUE UN NUMERO VALIDO (positivo mayor a 0)");
}
else{
    String inventarios = inventariotiempoTF.getText();
    String producciones = producciontiempoTF.getText();
    String empaquetados = empaquetadotiempoTF.getText();
    String salidas = salidatiempoTF.getText();

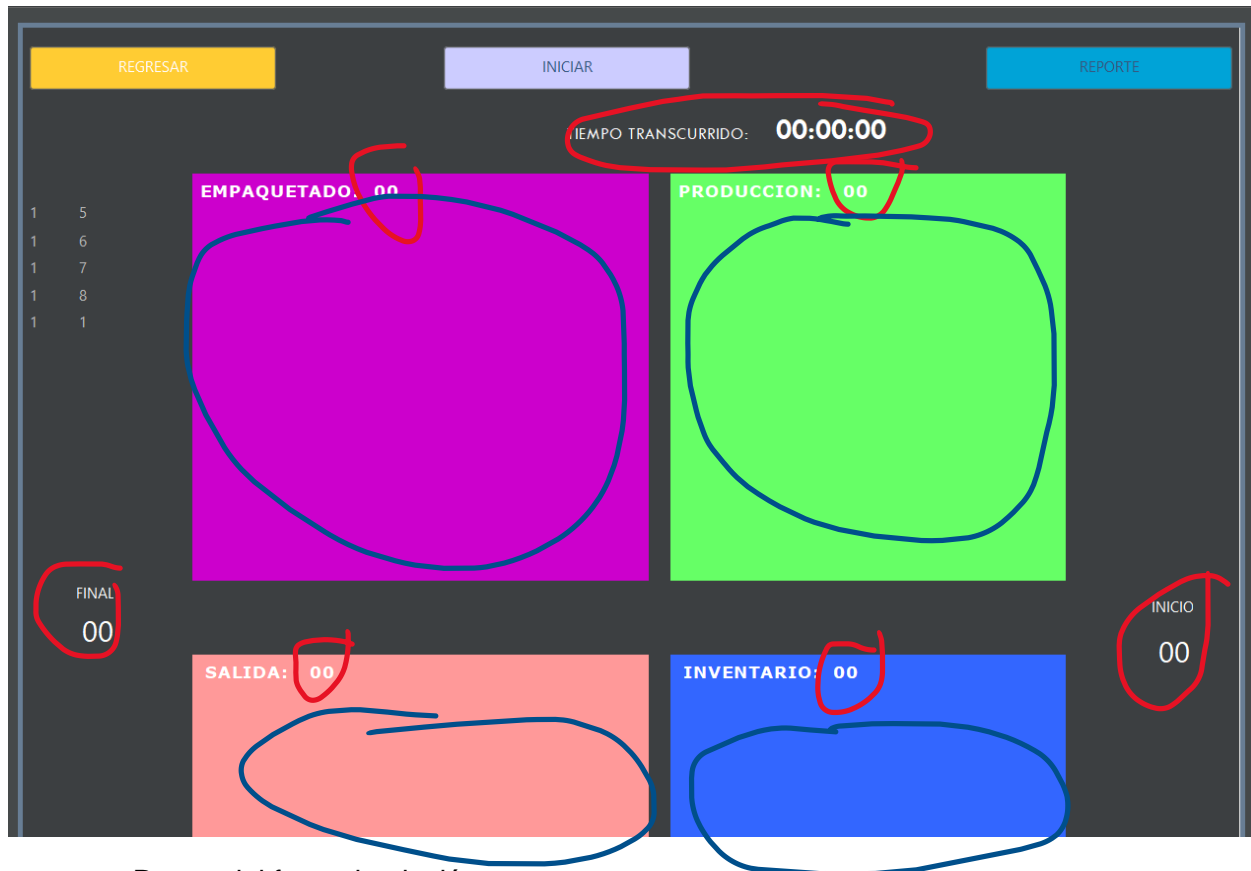
    String inventariosc = inventariocostoTF.getText();
    String produccionsc = produccioncostoTF.getText();
    String empaquetadosc = empaquetadocostoTF.getText();
    String salidasc = salidacostoTF.getText();

    inventariot = Integer.parseInt(inventarios);
    producciont = Integer.parseInt(producciones);
    empaquetadot = Integer.parseInt(empaquetados);
    salidat = Integer.parseInt(salidas);

    inventarioc = Double.parseDouble(inventariosc);
    produccionc = Double.parseDouble(produccionsc);
    empaquetadoc = Double.parseDouble(empaquetadosc);
    salidac = Double.parseDouble(salidasc);

    t1 = inventariot * inventarioc * 30;
    t2 = producciont * produccionc * 30;
    t3 = empaquetadot * empaquetadoc * 30;
    t4 = salidat * salidac * 30;
    t5 = t1 + t2 + t3 + t4;
}

```



Partes del form simulación:

- Lo circulado en rojo son contadores, muestran el resultado de valorar el tiempo e incrementar de unidad en unidad, los contadores tienen la capacidad de aumentar y disminuir según la cantidad.

```
public class HiloTemporizador implements Runnable {
    JLabel TiempoTranscurrido;
    boolean Temp=true;
    int horas, minutos, segundos;
    public HiloTemporizador(JLabel tiempoTranscurrido) {
        this.TiempoTranscurrido = tiempoTranscurrido;
    }
    public String toString(int numero) {
        if (numero < 10) {
            return "0" + Integer.toString(numero);
        }
        return Integer.toString(numero);
    }
    @Override
    public void run() {
        while (Temp) {
            try {
                Thread.sleep(1000);
            } catch (InterruptedException ex) {
                Logger.getLogger(simulacionFrame.class.getName()).log(Level.SEVERE, null, ex);
            }
            segundos++;
            if (segundos == 60) {minutos++;segundos = 0;}
            if (minutos == 60) {horas++;minutos = 0;}
            TiempoTranscurrido.setText(toString(horas) + ":" + toString(minutos) + ":" + toString(segundos));
            if (ContadorFinal==30) {
                JOptionPane.showMessageDialog(null, "TERMINADO, regrese para iniciar con nuevos tiempos o Imprima su reporte");
                Temp=false;
            }
        }
    }
}
```

- Para lo circulado en azul: son paneles que tienen la capacidad de almacenar los dibujos de los círculos hechos por un label, pero también sirven como delimitante visual según qué color (porción de código de lo que almacena un panel)

```
public class PintarCiculoInventario{  
    public PintarCiculoInventario(Graphics g, int x, int y) {  
        Color Inventario = new Color(0,0,102);  
        g.setColor(Inventario);  
        g.filloval(x, y, 20, 20);  
    }  
}
```

La metodología es simple, mientras el contador aumente, una clase crea un círculo y lo pinta de tal modo que concuerde con su panel, y después que el contador decremente los va despintando.

