

Prueba de Caja Blanca

“Título de proyecto: Sistema de inventario Medicina”

Integrantes:

Cesar Herrera

Kelly Montalvo

Matias Intriago

Fecha: 2025/12/03

Prueba caja blanca

RF N3

1. CÓDIGO FUENTE

Validación de stock y alarmas

```
public class SistemaCategorizacion {

    public static List<Alerta> verificar(Medicamento m, int diasAlerta) {

        if (m == null) {
            System.out.println("Medicamento inválido");
            return null;
        }

        List<Alerta> alertas = new ArrayList<>();

        if (m.stock <= m.stockMin)
            alertas.add(new Alerta("Stock bajo"));

        long dias = ChronoUnit.DAYS.between(LocalDate.now(), m.caducidad);

        if (dias >= 0 && dias <= diasAlerta)
            alertas.add(new Alerta("Próximo a vencer"));

        if (dias < 0)
            alertas.add(new Alerta("Producto vencido"));

        if (alertas.isEmpty())
            System.out.println("Producto sin alertas | status = ok");
        else
            System.out.println("Alertas detectadas | status = alerta");

        return alertas;
    }

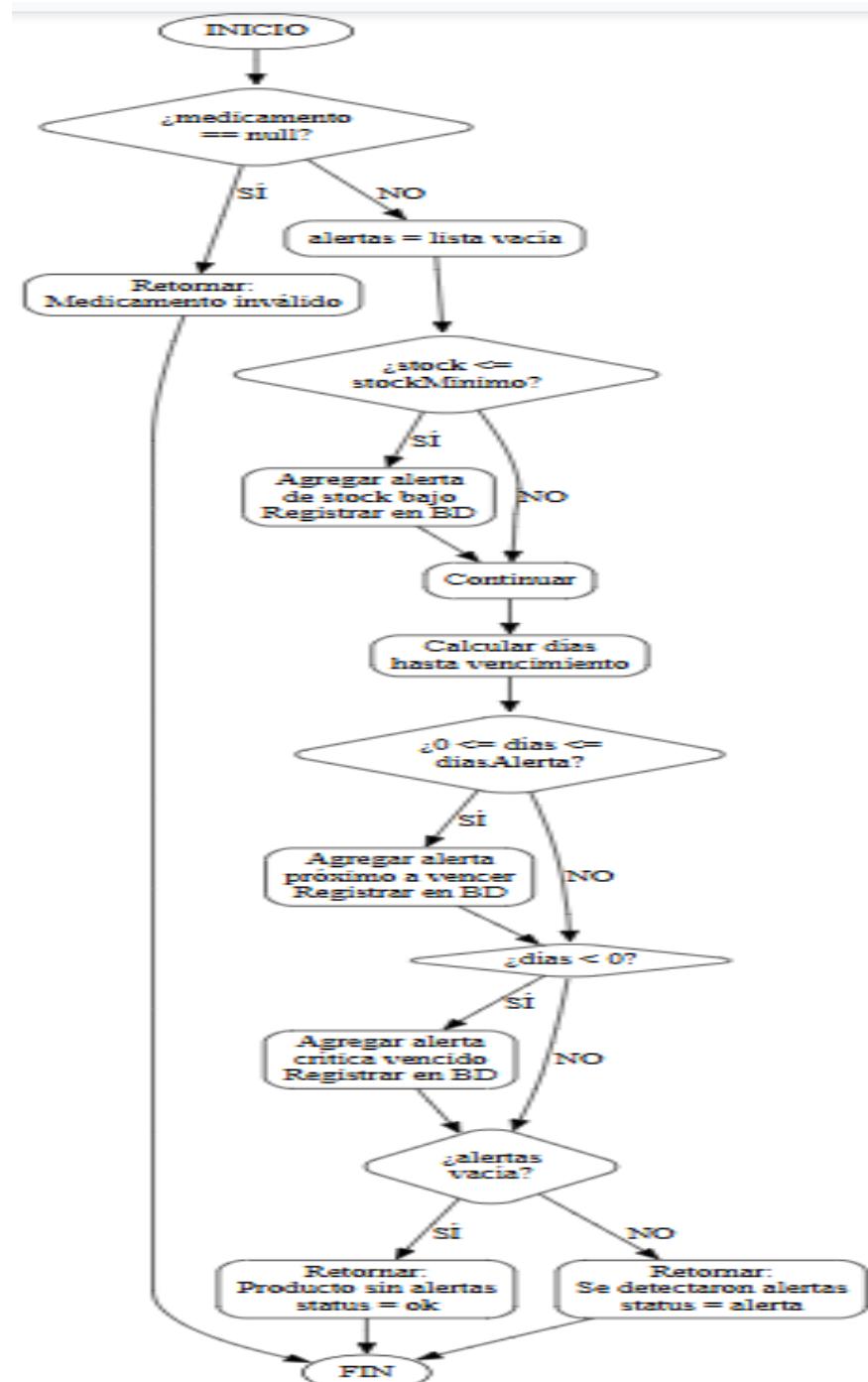
    public static void main(String[] args) {

        Medicamento m = new Medicamento(
                "Ibuprofeno", 5, 10,
                LocalDate.now().plusDays(3)
        );

        verificar(m, 7);
    }
}
```

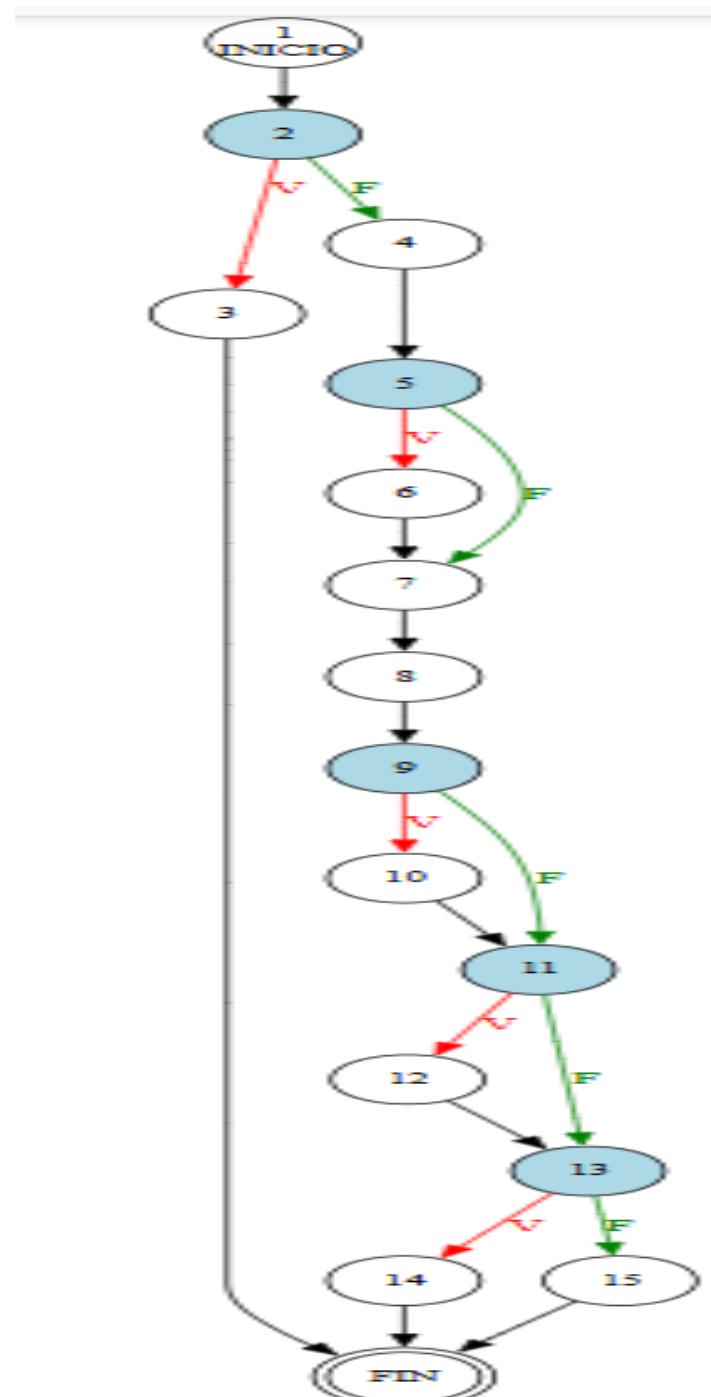
2. DIAGRAMA DE FLUJO (DF)

Diagrama de flujo del requisito 3



3. GRAFO DE FLUJO (GF)

Grafo de flujo del requisito 3



4. IDENTIFICACIÓN DE LAS RUTAS (Camino básico)

Determinar en base al GF del numeral 4

RUTAS

Ruta 1

Camino: 1 → 2 → 3 → FIN

Descripción: Medicamento nulo.

Datos de prueba:

medicamento = null

Ruta 2

Camino: 1 → 2 → 4 → 5 → 6 → 7 → 8 → 9 → 10 → 11 → 12 → 13 → 14 → FIN

Descripción: Stock bajo, medicamento próximo a vencer y además vencido (todas las alertas activadas).

Datos de prueba:

stock = 5

stockMin = 10

días = 15

vencido = true

Ruta 3

Camino: 1 → 2 → 4 → 5 → 7 → 8 → 9 → 11 → 13 → 14 → FIN

Descripción: Sin stock bajo, sin alerta de próximo vencimiento, sin alerta de vencimiento.

Datos de prueba:

stock = 50

stockMin = 10

días = 60

Ruta 4

Camino: 1 → 2 → 4 → 5 → 6 → 7 → 8 → 9 → 11 → 13 → 15 → FIN

Descripción: Solo alerta de stock bajo.

Datos de prueba:

stock = 5

stockMin = 10

días = 60

Ruta 5

Camino: 1 → 2 → 4 → 5 → 7 → 8 → 9 → 10 → 11 → 13 → 15 → FIN

Descripción: Solo alerta de próximo vencimiento.

Datos de prueba:

stock = 50

stockMin = 10

días = 20

Ruta 6

Camino: 1 → 2 → 4 → 5 → 7 → 8 → 9 → 11 → 12 → 13 → 15 → FIN

Descripción: Solo alerta de vencimiento.

Datos de prueba:

stock = 50

stockMin = 10

días = -10

5. COMPLEJIDAD CICLOMÁTICA

Se puede calcular de las siguientes formas:

Nodos (N): Son todos los círculos numerados.

Nodos predicados (P): Son los nodos de decisión, que tienen más de una salida: -

A (aristas): Contando todas las flechas entre nodos.

- $V(G) = \text{número de nodos predicados(decisiones)} + 1$

$$V(G) = P + 1$$

- $V(G) = A - N + 2$

DONDE

P: Número de nodos predicado

A: Número de aristas

N: Número de nodos

Datos del grafo de flujo

Nodos (N): 15

Nodos predicados (P): 5

Nodo 2: ¿medicamento == null?

Nodo 5: ¿stock ≤ stockMínimo?

Nodo 9: ¿0 ≤ días ≤ díasAlerta?

Nodo 11: ¿días < 0?

Nodo 13: ¿alertas vacía?

Aristas (A): Se identificaron inicialmente 18, pero el grafo real contiene 19.

Cálculo de la Complejidad Ciclomática

Método 1: $V(G) = P + 1$

Suma de nodos de decisión más 1:

$$V(G) = 5 + 1 = 6$$

Método 2: $V(G) = A - N + 2$

Usando aristas y nodos:

$$V(G) = 18 - 15 + 2 = 5$$

Como hay una discrepancia, se ajusta el número correcto de aristas a 19:

$$V(G) = 19 - 15 + 2 = 6$$

Resultado final

La complejidad ciclomática del grafo es: 6