

# Prueba de Caja Blanca

---

*“Diseño de sistema de gestión de contratos para una plaza comercial”*

## **Integrantes:**

Gallardo Vega Santiago Jose,  
Pérez Díaz David Ismael,  
Zambrano Cajas Isabela Valentina.

**Fecha: 2025-07-22**

## Requisito funcional 6: Editar y eliminar contratos registrados

El administrador deberá poder modificar el contenido contratos y eliminar contratos mediante una opcion que aparecerá en cada contrato

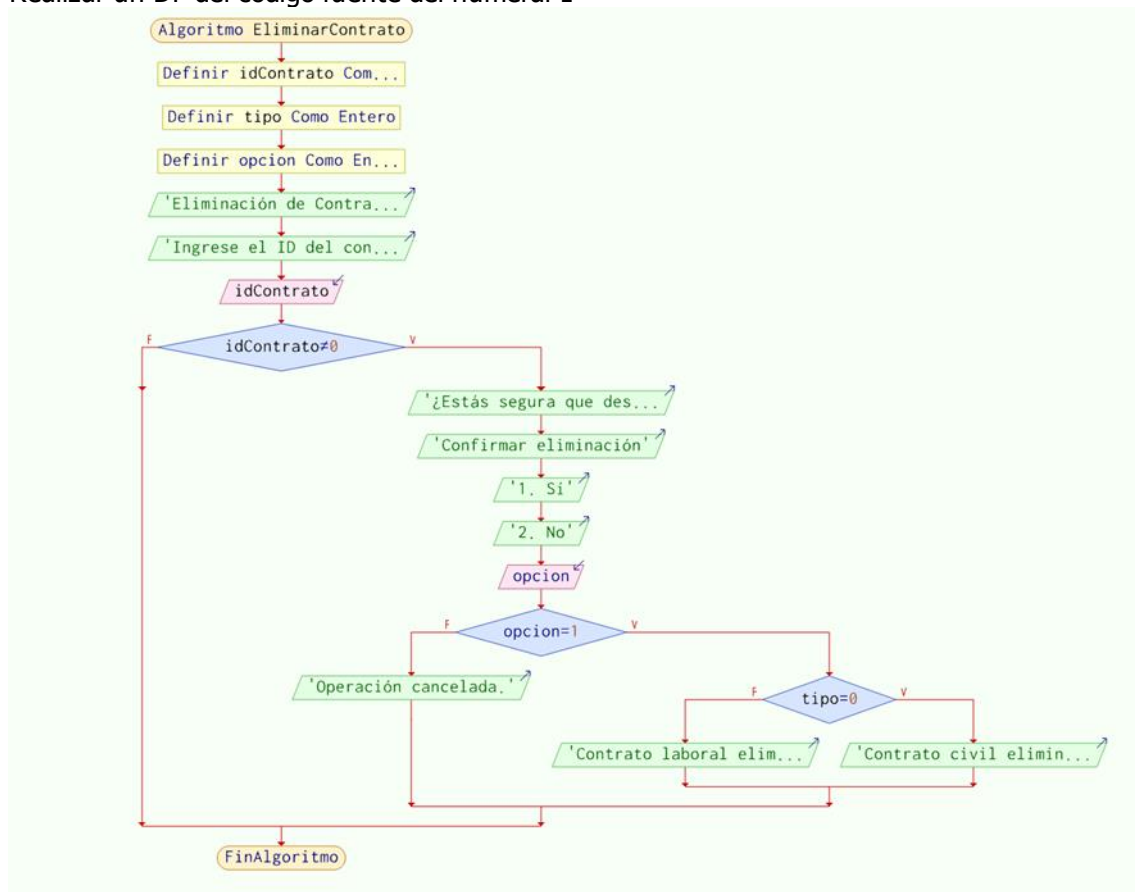
### 1. CÓDIGO FUENTE para eliminación de contratos

Pegar el trozo de código fuente que se requiere para el caso de prueba

```
public void eliminarContrato(ObjectId idContrato, int tipo) {  
    if (idContrato == null) {  
        JOptionPane.showMessageDialog(null,  
            "No se encontró el contrato para eliminar.",  
            "Error", JOptionPane.ERROR_MESSAGE);  
        return;  
    }  
  
    // 📌 Ventana de confirmación personalizada  
    int opcion = JOptionPane.showConfirmDialog(  
        null,  
        "¿Estás segura que deseas eliminar este contrato?\nEsta acción no se puede deshacer.",  
        "Confirmar eliminación",  
        JOptionPane.YES_NO_OPTION,  
        JOptionPane.WARNING_MESSAGE  
    );  
  
    if (opcion != JOptionPane.YES_OPTION) {  
        JOptionPane.showMessageDialog(null, "Operación cancelada.");  
        return;  
    }  
  
    try {  
        if (tipo == 0) {  
            MongoDBCivil servicio = new MongoDBCivil(userModel);  
            servicio.getCollection("ContratosCivil")  
                .deleteOne(new Document("_id", idContrato));  
            JOptionPane.showMessageDialog(null,  
                "✅ Contrato civil eliminado correctamente.");  
            formCivBus.dispose();  
        } else {  
            MongoDBCLaboral servicio = new MongoDBCLaboral(userModel);  
            servicio.getCollection("ContratosLaboral")  
                .deleteOne(new Document("_id", idContrato));  
            JOptionPane.showMessageDialog(null,  
                "✅ Contrato laboral eliminado correctamente.");  
            formLabBus.dispose();  
        }  
    } catch (Exception ex) {  
        JOptionPane.showMessageDialog(null,  
            "❌ Error al eliminar el contrato: " + ex.getMessage(),  
            "Error", JOptionPane.ERROR_MESSAGE);  
        ex.printStackTrace();  
    }  
}
```

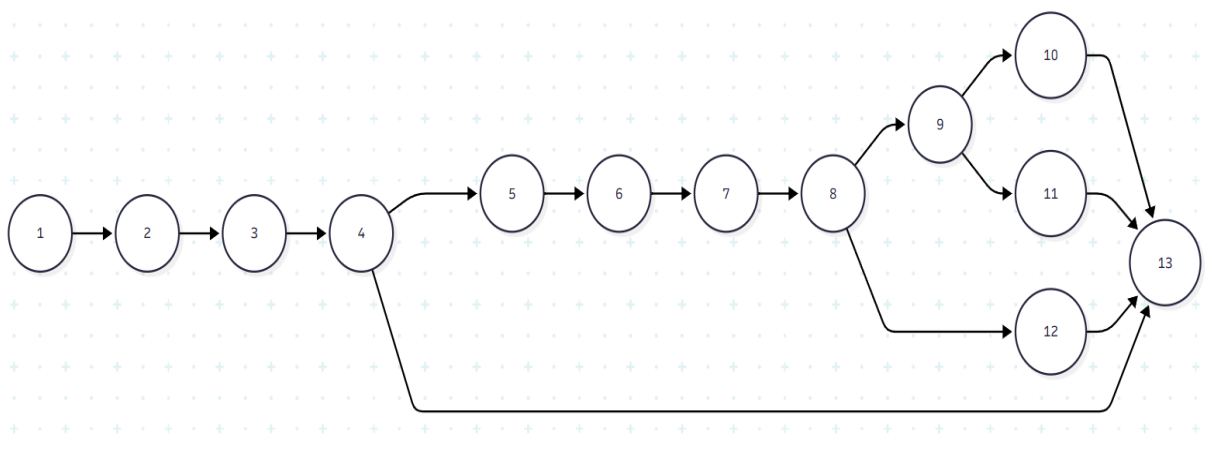
### 2. DIAGRAMA DE FLUJO (DF) eliminación de contratos

Realizar un DF del código fuente del numeral 1



### 3. GRAFO DE FLUJO (GF) eliminación de contratos

Realizar un GF en base al DF del numeral



### 4. IDENTIFICACIÓN DE LAS RUTAS (Camino básico) eliminación de contratos

Determinar en base al GF del numeral 4  
RUTAS

R1: 1-2-3-4-5-6-7-8-9-10-13

R2: 1-2-3-4-5-6-7-8-9-11-13

R3: 1-2-3-4-5-6-7-8-12-13

R4: 1-2-3-4-13

### **5. COMPLEJIDAD CICLOMÁTICA eliminación de contratos**

Se puede calcular de las siguientes formas:

$$V(G) = 3+1$$

$$V(G) = 4$$

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

$$V(G) = 15-13+2$$

$$V(G)=4$$

DONDE:

P: Número de nodos predicado

A: Número de aristas

N: Número de nodos

### **6. CÓDIGO FUENTE para edición de contratos**

Pegar el trozo de código fuente que se requiere para el caso de prueba

```

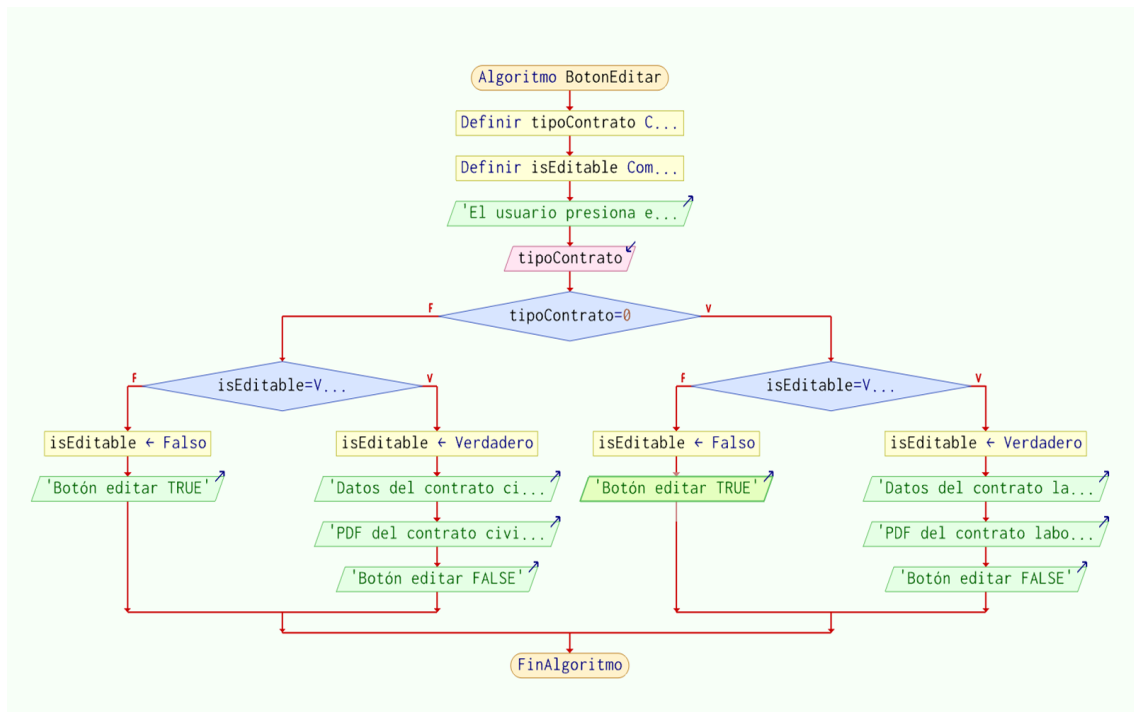
if (e.getSource() == formCivBus.BtnEditarCivil) {
    if(!isEditable){
        isEditable = setEditable(0,isEditable);
        System.out.println("Boton Editar TRUE");
    }
    else{
        this.mongoDBbusqueda.updateMongoDB(0, idCivilBuscado, obtenerTextosCampos(0));
        isEditable = setEditable(0,isEditable);
        try {
            MongoDBCcivil servicio = new MongoDBCcivil(userModel);
            new ContratoPdfGeneratoCiv().generarContratoPDF(servicio, idCivilBuscado);
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(null, "Error al generar el PDF civil.",
                "Error", JOptionPane.ERROR_MESSAGE);
            ex.printStackTrace();
        }
        System.out.println("Boton editar FALSE");
    }
}

if (e.getSource() == formLabBus.BtnEditarLab) {
    if(!isEditable){
        isEditable = setEditable(1,isEditable);
    }
    else{
        this.mongoDBbusqueda.updateMongoDB(1, idLaboralBuscado, obtenerTextosCampos(1));
        isEditable = setEditable(1,isEditable);
        try {
            MongoDBCLaboral servicio = new MongoDBCLaboral(userModel);
            new ContratoPdfGeneratoLab().generarContratoPDF(servicio, idLaboralBuscado);
        } catch (Exception ex) {
            JOptionPane.showMessageDialog(null, "Error al generar el PDF laboral.",
                "Error", JOptionPane.ERROR_MESSAGE);
            ex.printStackTrace();
        }
    }
}
}

```

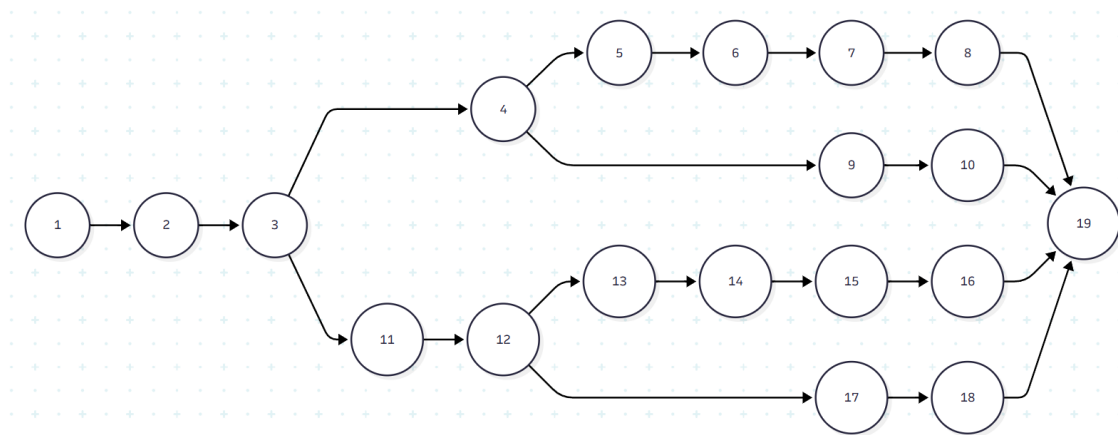
## 7. DIAGRAMA DE FLUJO (DF) edición de contratos

Realizar un DF del código fuente del numeral 1



## 8. GRAFO DE FLUJO (GF) edición de contratos

Realizar un GF en base al DF del numeral 4



## 9. IDENTIFICACIÓN DE LAS RUTAS (Camino básico) edición de contratos

Determinar en base al GF del numeral 4  
RUTAS

R1: 1-2-3-4-5-6-7-8-9-19  
R2: 1-2-3-4-9-10-19  
R3: 1-2-3-11-12-13-14-15-16-19  
R4: 1-2-3-11-12-17-18-19

### **10. COMPLEJIDAD CICLOMÁTICA edición de contratos**

Se puede calcular de las siguientes formas:

$$V(G) = 3+1$$

$$V(G) = 4$$

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

$$V(G) = 21-19+2$$

$$V(G) = 4$$

DONDE:

P: Número de nodos prediado

A: Número de aristas

N: Número de nodos