

Prueba de Caja Blanca

*“Implementación de un sistema de listado de servicios
para la empresa Muebles M&L”*

Integrantes:

Mateo Unda
Bernardo Suarez
Lizandro Rivera

Fecha 2025-01-16

Prueba caja blanca de describa el requisito funcional

1. CÓDIGO FUENTE

```
void modificarCatalogoYServicios() {
    int opcion;
    do {
        cout << "\n=== Modificar Catalogo y Servicios ===\n";
        cout << "1. Agregar al catalogo\n";
        cout << "2. Eliminar del catalogo\n";
        cout << "3. Agregar a servicios\n";
        cout << "4. Eliminar de servicios\n";
        cout << "5. Volver al menu principal\n";
        cout << "Seleccione una opcion: ";
        cin >> opcion;
        cin.ignore(); // Limpiar el buffer

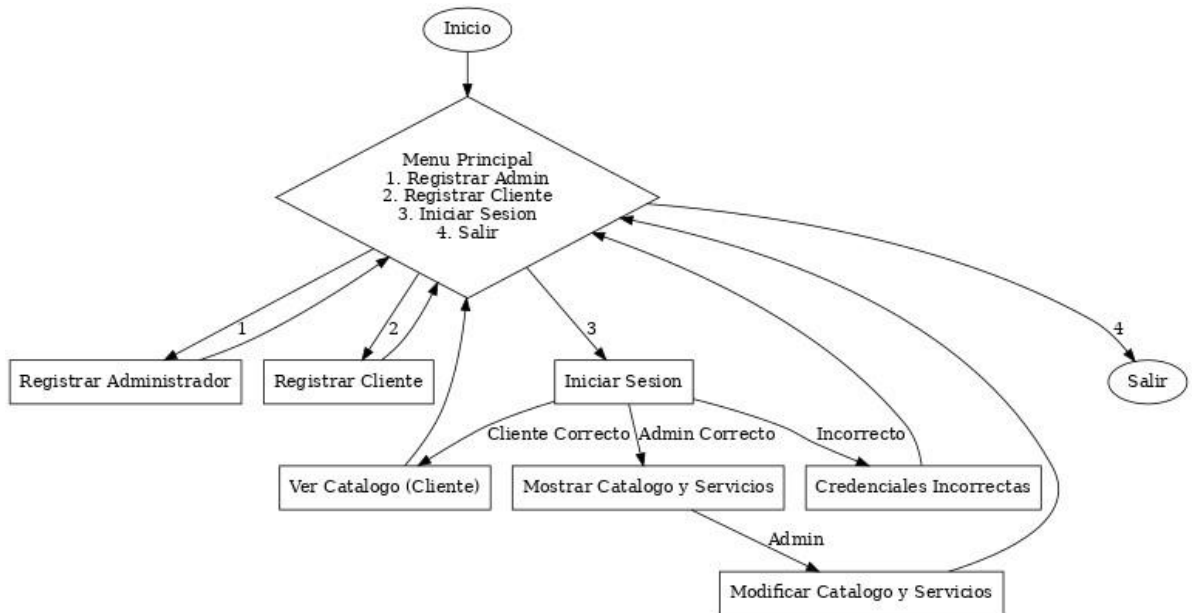
        if (opcion == 1) {
            if (catalogo.size() >= 10) {
                cout << "El catalogo esta lleno.\n";
                continue;
            }
            cout << "Ingrese el nombre del nuevo articulo: ";
            string articulo;
            getline(cin, articulo);
            catalogo.push_back(articulo);
            cout << "Articulo agregado exitosamente.\n";
        } else if (opcion == 2) {
            cout << "Ingrese el numero del articulo a eliminar: ";
            int eliminar;
            cin >> eliminar;
            cin.ignore();
            if (eliminar < 1 || eliminar > (int)catalogo.size()) {
                cout << "Opcion no valida.\n";
                continue;
            }
            catalogo.erase(catalogo.begin() + eliminar - 1);
            cout << "Articulo eliminado exitosamente.\n";
        } else if (opcion == 3) {
            if (servicios.size() >= 10) {
                cout << "La lista de servicios esta llena.\n";
                continue;
            }
            cout << "Ingrese el nombre del nuevo servicio: ";
            string servicio;
            getline(cin, servicio);
            servicios.push_back(servicio);
            cout << "Servicio agregado exitosamente.\n";
        } else if (opcion == 4) {
            cout << "Ingrese el numero del servicio a eliminar: ";
            int eliminar;
            cin >> eliminar;
            cin.ignore();
            if (eliminar < 1 || eliminar > (int)servicios.size()) {
                cout << "Opcion no valida.\n";
                continue;
            }
            servicios.erase(servicios.begin() + eliminar - 1);
            cout << "Servicio eliminado exitosamente.\n";
        } else if (opcion != 5) {
            cout << "Opcion no valida. Intente nuevamente.\n";
        }
    } while (opcion != 5);
}
```

```

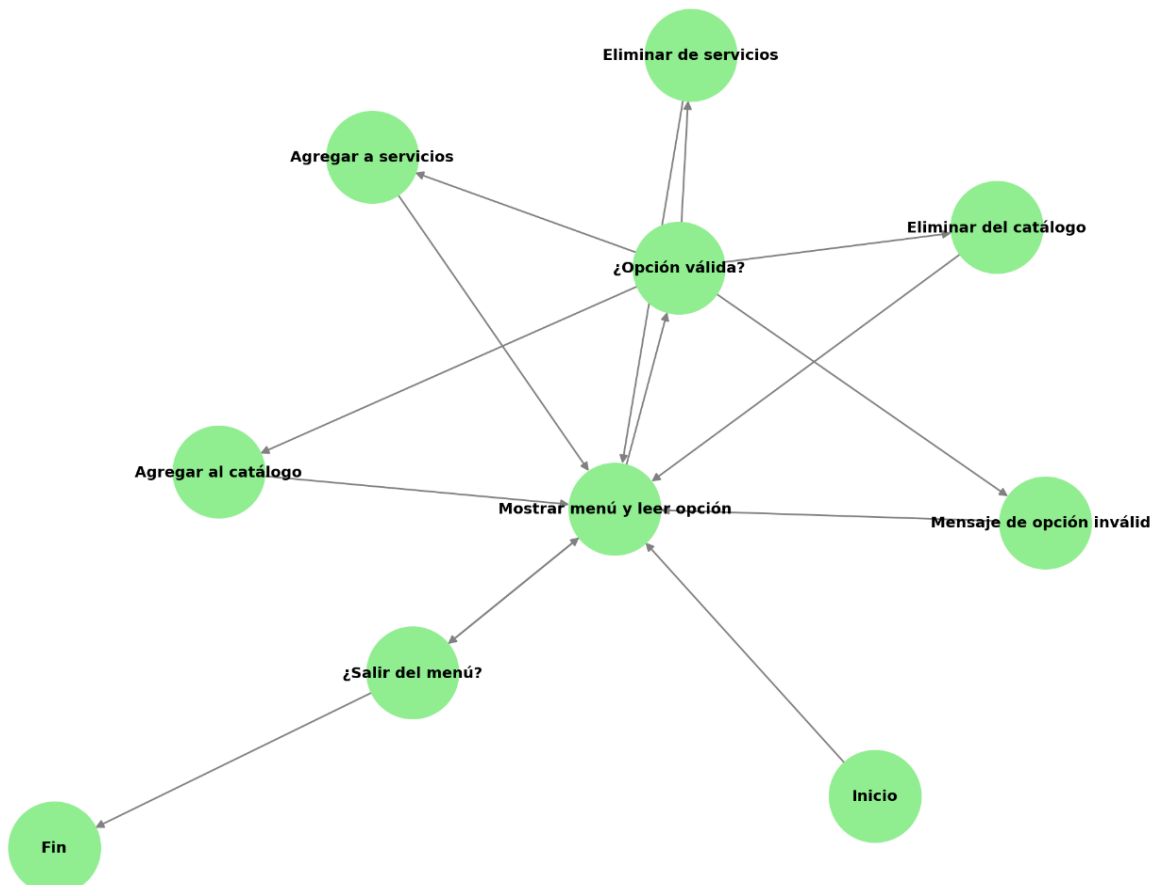
    }
  } while (opcion != 5);
}

```

2. DIAGRAMA DE FLUJO (DF)



3. GRAFO DE FLUJO (GF)



4. IDENTIFICACIÓN DE LAS RUTAS (Camino basico)

Determinar en base al GF del numeral 4

RUTAS

- **R1: 1 → 2 → 3 → 4 → 2 → 9 → 10**
- **R2: 1 → 2 → 3 → 5 → 2 → 9 → 10**
- **R3: 1 → 2 → 3 → 6 → 2 → 9 → 10**
- **R4: 1 → 2 → 3 → 7 → 2 → 9 → 10**
- **R5: 1 → 2 → 3 → 8 → 2 → 9 → 10**

5. COMPLEJIDAD CICLOMÁTICA

Método 1: Nodos predicados + 1

- Número de nodos predicados: 2
 - Nodo 3: Verificación de opción válida.
 - Nodo 9: Verificación de salida.
- **$V(G) = P + 1 = 2 + 1 = 3$**

Método 2: Aristas - Nodos + 2

- Número de aristas (A): 13
- Número de nodos (N): 10
- **$V(G) = A - N + 2 = 13 - 10 + 2 = 5$**

Ambos métodos confirman que la **complejidad ciclomática es 5**.