

Prueba de Caja Blanca

“Generación de proformas MarcallTex”

Integrantes:

Cañola Kevin

Marcalla

Cristhian

Lugamaña

Mateo

Tasiguano

Eduardo

Fecha: 2025/11/25

CONTROL DE VERSIONAMIENTO DE PRUEBAS CB

Version	Fecha	Responsable	Aprobado por
PCB_V1.0.0.docx			

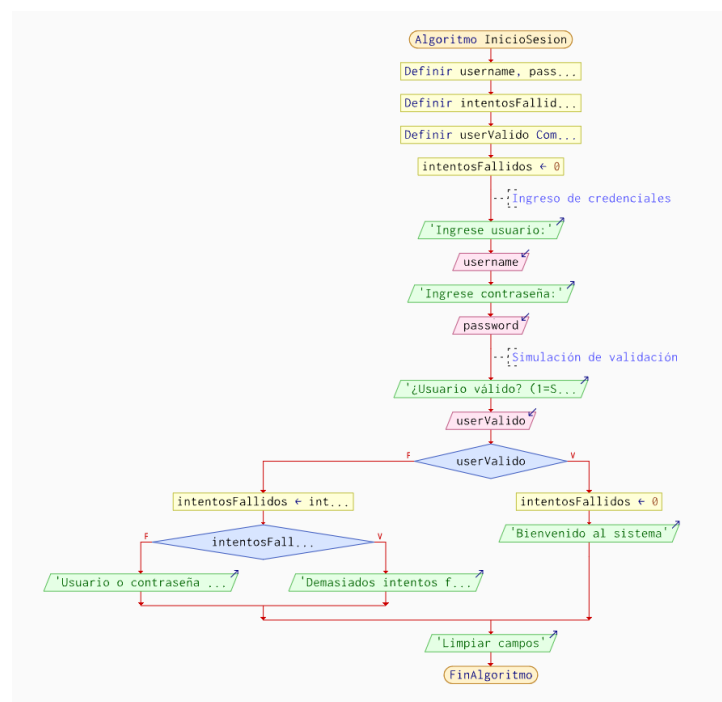
Prueba caja blanca

RF N1ª INICIO DE SESIÓN

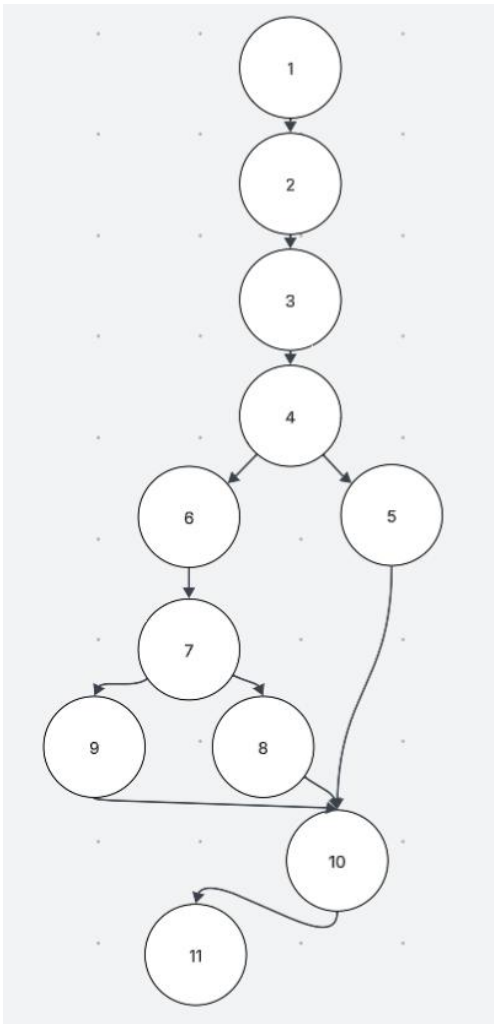
1. CÓDIGO FUENTE

```
35 Private Sub btnIngresar_Click(sender As Object, e As EventArgs) Handles btnIngresar.Click
36     Dim username As String = txtUsuario.Text.Trim()
37     Dim password As String = txtContraseña.Text.Trim()
38
39     Dim user = Usuario.ValidarLogin(username, password)
40
41     If user IsNot Nothing Then
42         intentosFallidos = 0
43
44         MsgBox("Bienvenido " & user.usuario, MsgBoxStyle.Information, "BIENVENIDO AL SISTEMA")
45
46         Dim admin As New ApartadoAdministrador()
47
48         ' ✓ flujo correcto
49         Me.Hide()
50         admin.Show()
51
52     Else
53         intentosFallidos += 1
54
55         If intentosFallidos >= 3 Then
56             MsgBox("Demasiados intentos fallidos.", MsgBoxStyle.Critical, "Error")
57             btnIngresar.Enabled = False
58             bloqueoTimer.Start()
59         Else
60             MsgBox("Usuario o contraseña incorrectos.", MsgBoxStyle.Critical, "Error")
61         End If
62     End If
63
64     LimpiarCampos()
65 End Sub
66
```

2. DIAGRAMA DE FLUJO (DF)



3. GRAFO DE FLUJO (GF)



4. IDENTIFICACIÓN DE LAS RUTAS (Camino básico)

Ruta	Secuencia de nodos	Descripción
R1	1 → 2 → 3 → 4 → 5 → 10 → 11	Usuario correcto, acceso concedido
R2	1 → 2 → 3 → 4 → 6 → 7 → 9 → 10 → 11	Credenciales incorrectas, menos de 3 intentos
R3	1 → 2 → 3 → 4 → 6 → 7 → 8 → 10 → 11	Credenciales incorrectas, tercer intento → bloqueo

5. COMPLEJIDAD CICLOMÁTICA

La complejidad ciclomática del Grafo de Flujo se calcula para determinar el número de caminos independientes del sistema.

Datos del grafo

- **N (Número de nodos):** 11
- **P (Número de nodos predicados):** 2
- **A (Número de aristas):** 12

Cálculo

Fórmula 1:

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

$$V(G) = 2 + 1 = 3$$

Fórmula 2:

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

$$V(G) = 12 - 11 + 2 = 3$$

Resultado

La complejidad ciclomática del requisito **Inicio de Sesión** es **3**, lo que indica que existen **tres caminos básicos independientes**.

DONDE

P: Número de nodos predicado

A: Número de aristas

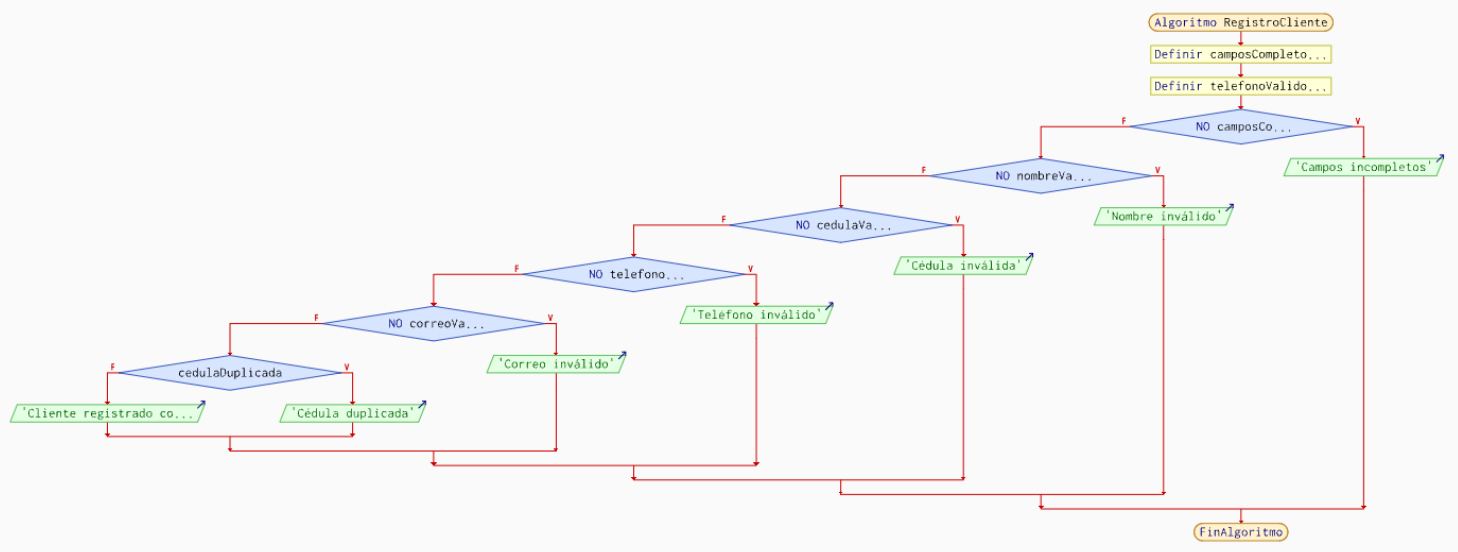
N: Número de nodos

RF N2ª REGISTRO DE CLIENTES

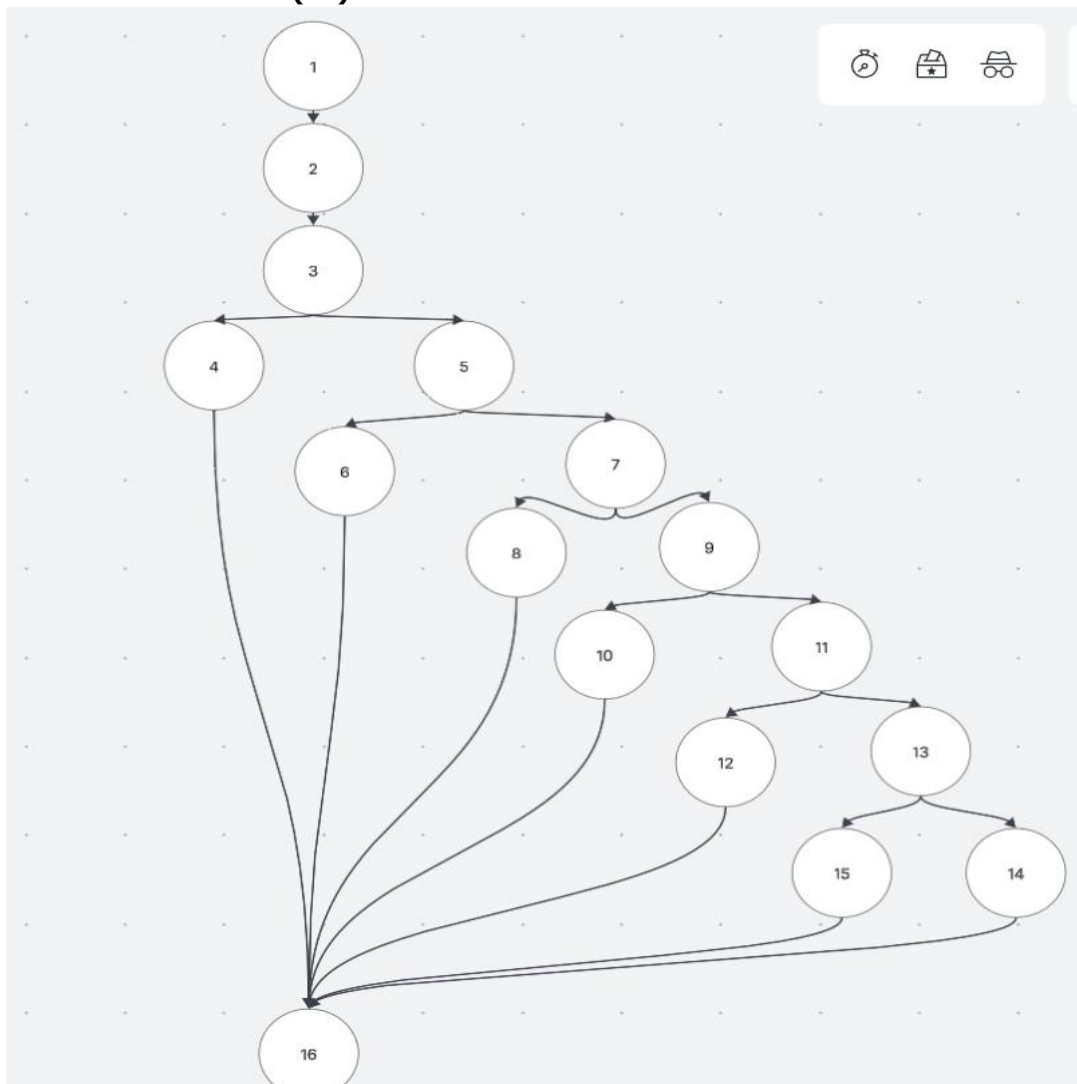
6. CÓDIGO FUENTE

```
1 Private Sub btnRegistrar_Click(sender As Object, e As EventArgs) Handles btnRegistrar.Click
2     Try
3         If txtNombre.Text.Trim() = "" OrElse
4             txtCedula.Text.Trim() = "" OrElse
5             txtTelefono.Text.Trim() = "" OrElse
6             txtCorreo.Text.Trim() = "" OrElse
7             txtDireccion.Text.Trim() = "" Then
8             MessageBox.Show("Por favor, complete todos los campos.")
9             Exit Sub
10        End If
11
12        If Not Regex.IsMatch(txtNombre.Text.Trim(), "[A-Za-zÁÉÍÓÚáéíóúÑñ ]+$") Then
13            MessageBox.Show("El nombre solo puede contener letras.")
14            Exit Sub
15        End If
16
17        If Not CedulaValida(txtCedula.Text.Trim()) Then
18            MessageBox.Show("La cédula ingresada no es válida.")
19            Exit Sub
20        End If
21
22        If Not Regex.IsMatch(txtTelefono.Text.Trim(), "\d{10}$") Then
23            MessageBox.Show("El teléfono debe contener 10 dígitos.")
24            Exit Sub
25        End If
26
27        If Not Regex.IsMatch(txtCorreo.Text.Trim(), "[a-z0-9._%+-]+@(gmail|outlook|hotmail|yahoo)\.(com|es)$") Then
28            MessageBox.Show("Ingrese un correo válido.")
29            Exit Sub
30        End If
31
32        Dim existente = coleccion.Find(filtroCedula).FirstOrDefault()
33        If existente IsNot Nothing Then
34            MessageBox.Show("Ya existe un cliente registrado con esta cédula.")
35            Exit Sub
36        End If
37
38        coleccion.InsertOne(nuevoCliente)
39        MessageBox.Show("Cliente registrado correctamente.")
40        LimpiarCampos()
41
42        Catch ex As Exception
43            MessageBox.Show("Error al registrar cliente.")
44        End Try
45    End Sub
```

7. DIAGRAMA DE FLUJO (DF)



8. GRAFO DE FLUJO (GF)



9. IDENTIFICACIÓN DE LAS RUTAS (Camino básico)

Ruta	Secuencia de nodos	Descripción
R1	1 → 2 → 3 → 5 → 7 → 9 → 11 → 13 → 15 → 16	Registro exitoso del cliente
R2	1 → 2 → 3 → 4 → 16	Campos incompletos
R3	1 → 2 → 3 → 5 → 6 → 16	Nombre inválido
R4	1 → 2 → 3 → 5 → 7 → 8 → 16	Cédula inválida
R5	1 → 2 → 3 → 5 → 7 → 9 → 10 → 16	Teléfono inválido
R6	1 → 2 → 3 → 5 → 7 → 9 → 11 → 12 → 16	Correo inválido
R7	1 → 2 → 3 → 5 → 7 → 9 → 11 → 13 → 14 → 16	Cédula duplicada

10. COMPLEJIDAD CICLOMÁTICA

La complejidad ciclomática del Grafo de Flujo se calcula para determinar el número de caminos independientes del sistema.

Datos del grafo

- **N (Número de nodos):** 16
- **P (Número de nodos predicados):** 6
- **A (Número de aristas):** 21

Cálculo

Fórmula 1:

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

$$V(G) = 6 + 1 = 7$$

Fórmula 2:

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

$$V(G) = 21 - 16 + 2 = 7$$

Resultado

La complejidad ciclomática del requisito **Registro de Clientes** es **7**, lo que indica que existen **siete caminos básicos independientes**.

DONDE

P: Número de nodos predicado

A: Número de aristas

N: Número de nodos