

# Prueba de Caja Blanca

---

## “ Sistema Automatizado de Gestión de Historias Clínicas para el Centro Veterinario AniMedical ”

### Integrantes:

Paul Chillagana  
Andres Pallango  
Jorge Nasimba

Fecha 18/02/2024

### BUSCAR PROBLEMAS

#### Prueba caja blanca validar cedula

#### CÓDIGO FUENTE

```
<script>

function validateForm() {

    var cedula = document.getElementById("cedula_d").value;

    // Preguntamos si la cedula consta de 10 digitos

    if (cedula.length == 10) {

        // Obtenemos el digito de la region que son los dos primeros
digitos

        var digito_region = cedula.substring(0, 2);

        // Pregunto si la region existe, Ecuador se divide en 24 regiones
```

```
if (digito_region >= 1 && digito_region <= 24) {

    // Extraigo el ultimo digito

    var ultimo_digito = cedula.substring(9, 10);

    // Agrupo todos los pares y los sumo

    var pares = parseInt(cedula.substring(1, 2)) +
parseInt(cedula.substring(3, 4)) +

        parseInt(cedula.substring(5, 6)) +
parseInt(cedula.substring(7, 8));

    // Agrupo los impares, los multiplico por un factor de 2,

    // si la resultante es > que 9 le restamos el 9 a la resultante

    var impares = 0;

    for (var i = 0; i < 9; i += 2) {

        var numero = parseInt(cedula.substring(i, i + 1)) * 2;

        impares += (numero > 9) ? (numero - 9) : numero;

    }

    // Suma total

    var suma_total = pares + impares;

    // Extraemos el primer digito

    var primer_digito_suma = String(suma_total).substring(0, 1);
```

```
    // Obtenemos la decena inmediata

    var decena = (parseInt(primer_digito_suma) + 1) * 10;

    // Obtenemos la resta de la decena inmediata - la suma_total

    // Esto nos da el digito validador

    var digito_validador = decena - suma_total;

    // Si el digito validador es igual a 10, toma el valor de 0

    if (digito_validador == 10)

        digito_validador = 0;
```

```

        // Validamos que el digito validador sea igual al de la cedula

        if (digito_validador == ultimo_digito) {

            alert("Cédula correcta");

            return true;

        } else {

            alert("Cédula incorrecta");

            return false;

        }

    } else {

        alert("Esta cedula no pertenece a ninguna región");

        return false;

    }

} else {

    alert("Esta cedula tiene menos de 10 dígitos");

    return false;

}

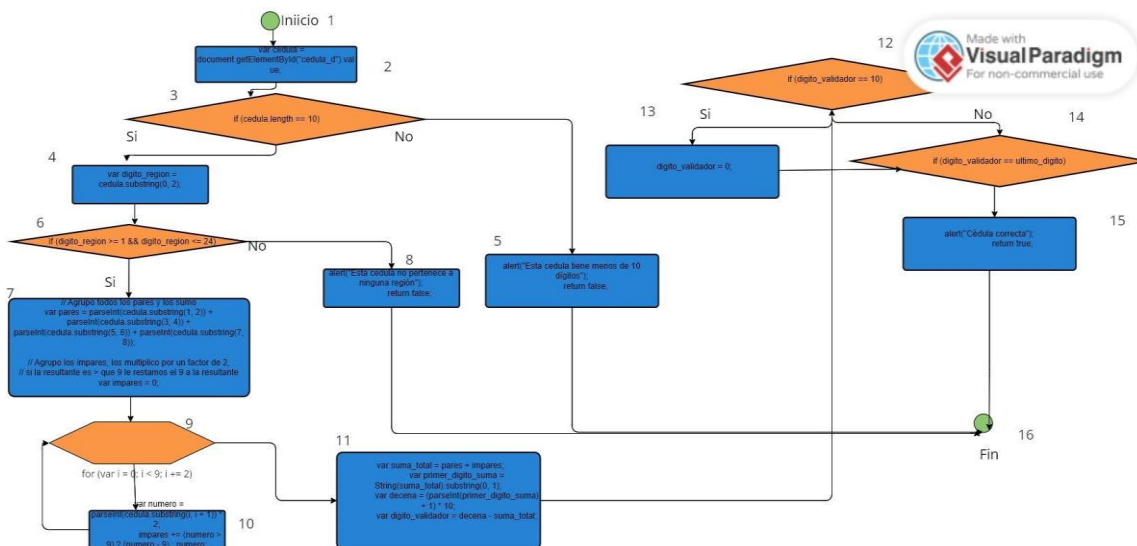
}

}

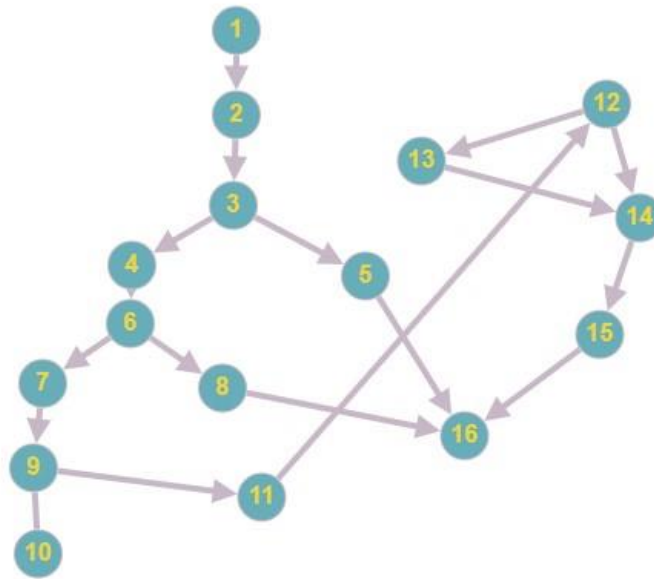
</script>

```

## DIAGRAMA DE FLUJO



## Grafo



## RUTAS

**Ruta1: 1,2,3,5,16**

**Ruta2: 1,2,3,4,6,8,16**

**Ruta3: 1,2,3,4,6,7,9,10,9,11,12,13,14,15,16**

**Ruta4: 1,2,3,4,6,7,9,10,9,11,12,14,15,16**

## COMPLEJIDAD CICLOMÁTICA

$$V(G)=E-N+2P.$$

- $N=16$  (número de nodos)
- $E=17$  (número de aristas)
- $P=1$  (número de componentes conexos, ya que el grafo está completamente conectado)

Por lo tanto, la complejidad ciclomática es:

$$V(G)=17-16+2*1=17-16+2=3$$

## Prueba caja blanca NUEVA HISTORIA CLINICA

### CÓDIGO FUENTE

```
@app.route('/nueva_historia', methods=['GET', 'POST'])
def nueva_historia():
    error_message = None # Inicializa el mensaje de error como nulo

    if request.method == 'POST':
        cedula = request.form['cedula']
```

```
propietario = request.form['propietario']
direccion = request.form['direccion']
medico_responsable = request.form['medico_responsable']
fecha_creacion = request.form['fecha_creacion']
telefono = request.form['telefono']
nombre_paciente = request.form['nombre_paciente']
fecha_nacimiento = request.form['fecha_nacimiento']
especie = request.form['especie']
raza = request.form['raza']
sexo = request.form['sexo']
color = request.form['color']
vacuna_1 = request.form['vacuna_1']
fecha_vacuna_1 = request.form['fecha_vacuna_1'] if request.form['fecha_vacuna_1'] else
None
vacuna_2 = request.form['vacuna_2']
fecha_vacuna_2 = request.form['fecha_vacuna_2'] if request.form['fecha_vacuna_2'] else
None
vacuna_3 = request.form['vacuna_3']
fecha_vacuna_3 = request.form['fecha_vacuna_3'] if request.form['fecha_vacuna_3'] else
None
vacuna_4 = request.form['vacuna_4']
fecha_vacuna_4 = request.form['fecha_vacuna_4'] if request.form['fecha_vacuna_4'] else
None
vacuna_5 = request.form['vacuna_5']
fecha_vacuna_5 = request.form['fecha_vacuna_5'] if request.form['fecha_vacuna_5'] else
None
fecha_ultima_desparasitacion = request.form['fecha_ultima_desparasitacion'] if
request.form['fecha_ultima_desparasitacion'] else None
motivo_consulta = request.form['motivo_consulta']
sintomatologia = request.form['sintomatologia']
tratamiento = request.form['tratamiento']
diagnostico_diferencial = request.form['diagnostico_diferencial']
examenes_complementarios = request.form['examenes_complementarios']
diagnostico_definitivo = request.form['diagnostico_definitivo']
tratamiento_final = request.form['tratamiento_final']
```

```

medicamento_1 = request.form['medicamento_1']
posologia_1 = request.form['posologia_1']
medicamento_2 = request.form['medicamento_2']
posologia_2 = request.form['posologia_2']
medicamento_3 = request.form['medicamento_3']
posologia_3 = request.form['posologia_3']
medicamento_4 = request.form['medicamento_4']
posologia_4 = request.form['posologia_4']
medicamento_5 = request.form['medicamento_5']
posologia_5 = request.form['posologia_5']
proxima_cita = request.form['proxima_cita'] if request.form['proxima_cita'] else None

# Verifica que los campos obligatorios estén llenos
if not (cedula and propietario and direccion and medico_responsable and fecha_creacion
and telefono and
        nombre_paciente and especie and sexo):
    error_message = 'Por favor, llena todos los campos obligatorios.'
else:
    cursor = mysql.get_db().cursor()
    cursor.execute("INSERT INTO historia (cedula, propietario, direccion,
medico_responsable, fecha_creacion, telefono, nombre_paciente, fecha_nacimiento, especie,
raza, sexo, color, vacuna_1, fecha_vacuna_1, vacuna_2, fecha_vacuna_2, vacuna_3,
fecha_vacuna_3, vacuna_4, fecha_vacuna_4, vacuna_5, fecha_vacuna_5,
fecha_ultima_desparasitacion, motivo_consulta, sintomatologia, tratamiento,
diagnostico_diferencial, exámenes_complementarios, diagnostico_definitivo,
tratamiento_final, medicamento_1, posologia_1, medicamento_2, posologia_2,
medicamento_3, posologia_3, medicamento_4, posologia_4, medicamento_5, posologia_5,
proxima_cita) VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s,
%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)",
        (cedula, propietario, direccion, medico_responsable, fecha_creacion, telefono,
nombre_paciente, fecha_nacimiento, especie, raza, sexo, color, vacuna_1, fecha_vacuna_1,
vacuna_2, fecha_vacuna_2, vacuna_3, fecha_vacuna_3, vacuna_4, fecha_vacuna_4,
vacuna_5, fecha_vacuna_5, fecha_ultima_desparasitacion, motivo_consulta, sintomatologia,
tratamiento, diagnostico_diferencial, exámenes_complementarios, diagnostico_definitivo,
tratamiento_final, medicamento_1, posologia_1, medicamento_2, posologia_2,
medicamento_3, posologia_3, medicamento_4, posologia_4, medicamento_5, posologia_5,
proxima_cita))

```

```

mysql.get_db().commit()

cursor.close()

return redirect(url_for('historia_creada'))

return render_template('nueva_historia.html', error_message=error_message)

# Nueva ruta para historia clínica dermatológica
@app.route('/nueva_historia_d', methods=['GET', 'POST'])
def nueva_historia_d():
    error_message = None

    if request.method == 'POST':
        cedula_d = request.form['cedula_d']
        propietario_d = request.form['propietario_d']
        direccion_d = request.form['direccion_d']
        medico_responsable_d = request.form['medico_responsable_d']
        fecha_creacion_d = request.form['fecha_creacion_d']
        telefono_d = request.form['telefono_d']
        nombre_paciente_d = request.form['nombre_paciente_d']
        fecha_nacimiento_d = request.form['fecha_nacimiento_d']
        especie_d = request.form['especie_d']
        raza_d = request.form['raza_d']
        sexo_d = request.form['sexo_d']
        color_d = request.form['color_d']
        vacuna_1_d = request.form['vacuna_1_d']
        fecha_vacuna_1_d = request.form['fecha_vacuna_1_d'] if
request.form['fecha_vacuna_1_d'] else None
        vacuna_2_d = request.form['vacuna_2_d']
        fecha_vacuna_2_d = request.form['fecha_vacuna_2_d'] if
request.form['fecha_vacuna_2_d'] else None
        vacuna_3_d = request.form['vacuna_3_d']

```

```
    fecha_vacuna_3_d = request.form['fecha_vacuna_3_d'] if
request.form['fecha_vacuna_3_d'] else None

    vacuna_4_d = request.form['vacuna_4_d']

    fecha_vacuna_4_d = request.form['fecha_vacuna_4_d'] if
request.form['fecha_vacuna_4_d'] else None

    vacuna_5_d = request.form['vacuna_5_d']

    fecha_vacuna_5_d = request.form['fecha_vacuna_5_d'] if
request.form['fecha_vacuna_5_d'] else None

    fecha_ultima_desparasitacion_d = request.form['fecha_ultima_desparasitacion_d'] if
request.form['fecha_ultima_desparasitacion_d'] else None

    motivo_consulta_d = request.form['motivo_consulta_d']

    sintomatologia_d = request.form['sintomatologia_d']

    tratamiento_d = request.form['tratamiento_d']

    diagnostico_diferencial_d = request.form['diagnostico_diferencial_d']

    otras_mascotas_d = request.form['otras_mascotas_d']

    nin_casa_d = request.form['nin_casa_d']

    familia_problema_d = request.form['familia_problema_d']

    tipo_comida_d = request.form['tipo_comida_d']

    golosinas_d = request.form['golosinas_d']

    caida_pelo_d = request.form['caida_pelo_d']

    se_rasca_d = request.form['se_rasca_d']

    ambiente_d = request.form['ambiente_d']

    pasa_sol_d = request.form['pasa_sol_d']

    pasa_tierra_d = request.form['pasa_tierra_d']

    defecacion_d = request.form['defecacion_d']

    parte_enrojecida_d = request.form['parte_enrojecida_d']

    fecha_ectoparasitos_d = request.form['fecha_ectoparasitos_d'] if
request.form['fecha_ectoparasitos_d'] else None

    descrip_ectoparasitos_d = request.form['descrip_ectoparasitos_d']

    duchas_casa_d = request.form['duchas_casa_d']

    alergia_comida_d = request.form['alergia_comida_d']

    rasp_cutaneo_d = request.form['rasp_cutaneo_d']

    tricograma_d = request.form['tricograma_d']

    lampara_wood_d = request.form['lampara_wood_d']
```



```

reflejo_otopodal_d = request.form['reflejo_otopodal_d']

biopsia_d = request.form['biopsia_d']

citologia_d = request.form['citologia_d']

antibiograma_d = request.form['antibiograma_d']


diagnostico_definitivo_d = request.form['diagnostico_definitivo_d']

tratamiento_final_d = request.form['tratamiento_final_d']

medicamento_1_d = request.form['medicamento_1_d']

posologia_medicamento_1_d = request.form['posologia_medicamento_1_d']

medicamento_2_d = request.form['medicamento_2_d']

posologia_medicamento_2_d = request.form['posologia_medicamento_2_d']

medicamento_3_d = request.form['medicamento_3_d']

posologia_medicamento_3_d = request.form['posologia_medicamento_3_d']

medicamento_4_d = request.form['medicamento_4_d']

posologia_medicamento_4_d = request.form['posologia_medicamento_4_d']

medicamento_5_d = request.form['medicamento_5_d']

posologia_medicamento_5_d = request.form['posologia_medicamento_5_d']

proxima_cita_d = request.form['proxima_cita_d'] if request.form['proxima_cita_d'] else
None


# Verifica que los campos obligatorios estén llenos

if not (cedula_d and propietario_d and direccion_d and medico_responsable_d and
fecha_creacion_d and telefono_d and

    nombre_paciente_d and especie_d and sexo_d):

    error_message = 'Por favor, llena todos los campos obligatorios.'

else:

    cursor = mysql.get_db().cursor()

    cursor.execute("INSERT INTO historia_derma (cedula_d, propietario_d, direccion_d,
medico_responsable_d, fecha_creacion_d, telefono_d, nombre_paciente_d,
fecha_nacimiento_d, especie_d, raza_d, sexo_d, color_d, vacuna_1_d, fecha_vacuna_1_d,
vacuna_2_d, fecha_vacuna_2_d, vacuna_3_d, fecha_vacuna_3_d, vacuna_4_d,
fecha_vacuna_4_d, vacuna_5_d, fecha_vacuna_5_d, fecha_ultima_desparasitacion_d,
motivo_consulta_d, sintomatologia_d, tratamiento_d, diagnostico_diferencial_d,
otras_mascotas_d, nin_casa_d, familia_problema_d, tipo_comida_d,
golosinas_d,caida_pelo_d, se_rasca_d, ambiente_d, pasa_sol_d, pasa_tierra_d,
defecacion_d,parte_enrojecida_d, fecha_ectoparasitos_d, descrip_ectoparasitos_d,
duchas_casa_d, alergia_comida_d, rasp_cutaneo_d, tricograma_d, lampara_wood_d,

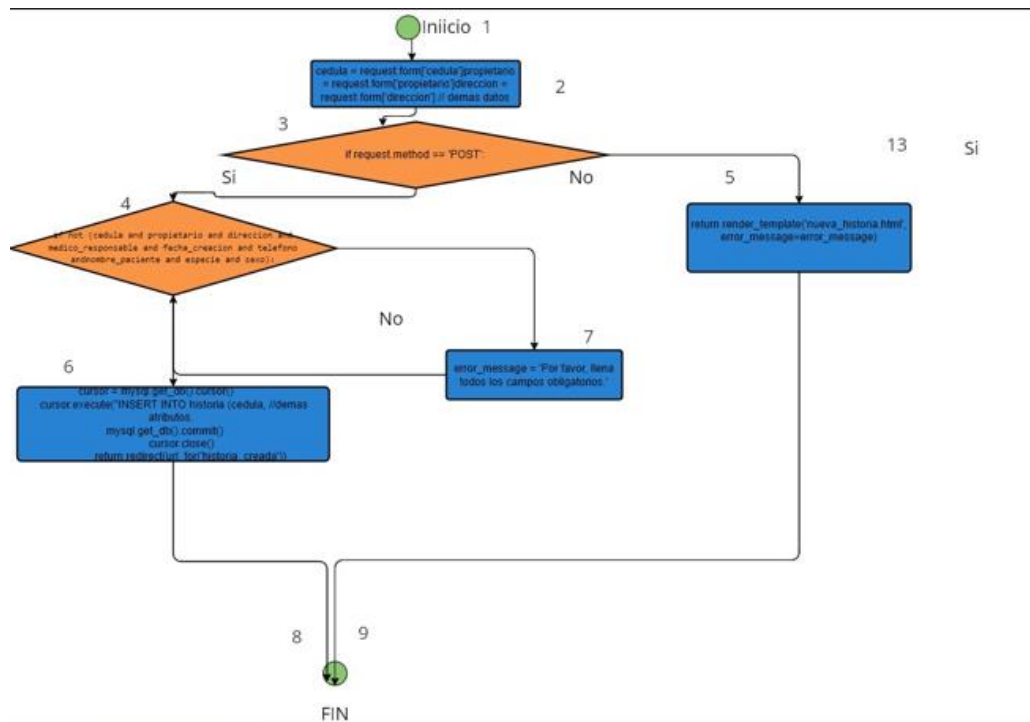
```

```
(cedula_d, propietario_d, direccion_d, medico_responsable_d, fecha_creacion_d, telefono_d, nombre_paciente_d, fecha_nacimiento_d, especie_d, raza_d, sexo_d, color_d, vacuna_1_d, fecha_vacuna_1_d, vacuna_2_d, fecha_vacuna_2_d, vacuna_3_d, fecha_vacuna_3_d, vacuna_4_d, fecha_vacuna_4_d, vacuna_5_d, fecha_vacuna_5_d, fecha_ultima_desparasitacion_d, motivo_consulta_d, sintomatologia_d, tratamiento_d, diagnostico_diferencial_d, otras_mascotas_d, nin_casa_d, familia_problema_d, tipo_comida_d, golosinas_d, caida_pelo_d, se_rasca_d, ambiente_d, pasa_sol_d, pasa_tierra_d, defecacion_d, parte_enrojecida_d, fecha_ectoparasitos_d, descrip_ectoparasitos_d, duchas_casa_d, alergia_comida_d, rasp_cutaneo_d, tricograma_d, lampara_wood_d, reflejo_otopodal_d, biopsia_d, citologia_d, antibiograma_d, diagnostico_definitivo_d, tratamiento_final_d, medicamento_1_d, posologia_medicamento_1_d, medicamento_2_d, posologia_medicamento_2_d, medicamento_3_d, posologia_medicamento_3_d, medicamento_4_d, posologia_medicamento_4_d, medicamento_5_d, posologia_medicamento_5_d, proxima_cita_d))
```

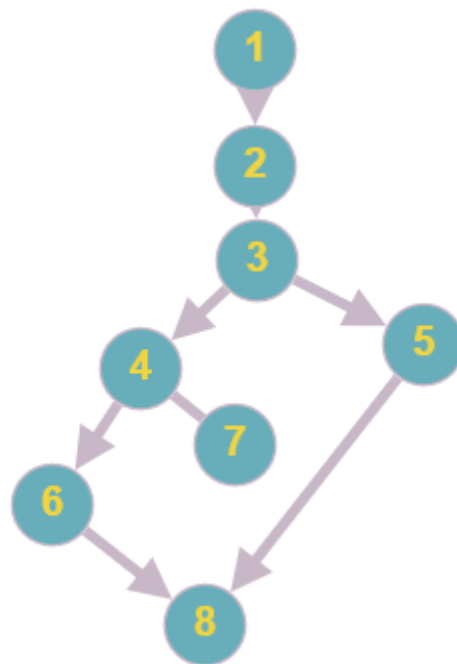
```
cursor.close()
```

```
return render_template('nueva_historia_d.html', error_message=error_message)
```

## DIAGRAMA DE FLUJO



## Grafo



## RUTAS

**Ruta1: 1,2,3,4,6,8**

**Ruta2: 1,2,3,4,7,4,6,8**

**Ruta3: 1,2,3,5,8**

## COMPLEJIDAD CICLOMÁTICA

$$V(G)=E-N+2P.$$

- Número de nodos (N) = 8
- Número de aristas (E) = 17
- Número de componentes conexos (P) = 1

Podemos calcular la complejidad ciclomática utilizando la fórmula:

$$V(G) = E - N + 2P$$

Sustituyendo los valores:

$$V(G) = 17 - 8 + 2(1) \quad V(G) = 17 - 8 + 2 \quad V(G) = 9 + 2 \quad V(G) = 11$$

## Prueba caja blanca buscar historia clinica

### CÓDIGO FUENTE

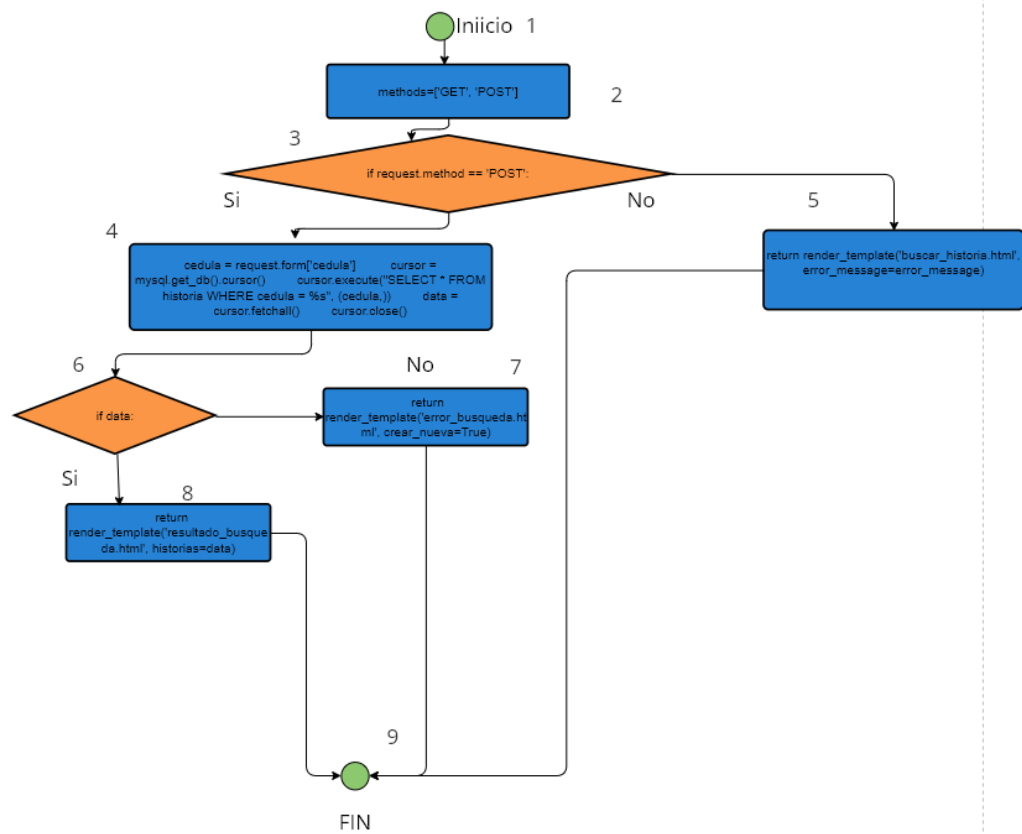
```
@app.route('/buscar_historia', methods=['GET', 'POST'])
def buscar_historia():
    if request.method == 'POST':
        cedula = request.form['cedula']

        cursor = mysql.get_db().cursor()
        cursor.execute("SELECT * FROM historia WHERE cedula = %s", (cedula,))
        data = cursor.fetchall()
        cursor.close()

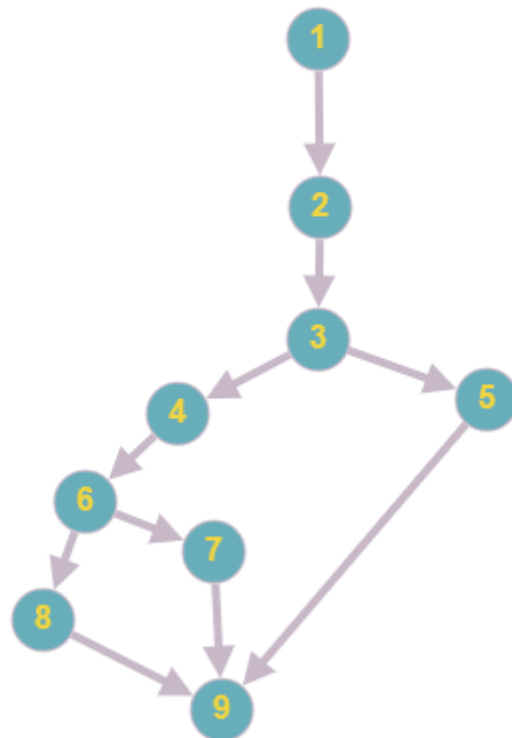
        if data:
            return render_template('resultado_busqueda.html', historias=data)
        else:
            return render_template('error_busqueda.html', crear_nueva=True)

    return render_template('buscar_historia.html')
```

## DIAGRAMA DE FLUJO



## Grafo



## **RUTAS**

**Ruta1: 1,2,3,5,9**

**Ruta2: 1,2,3,4,6,7,9**

**Ruta3: 1,2,3,4,6,8,9,**

## **COMPLEJIDAD CICLOMÁTICA**

- *Número de nodos (N): 9*
- *Número de aristas (E): 16*
- *Número de componentes conexos (P): 1 (el grafo está completamente conectado)*

*Podemos calcular la complejidad ciclomática:*

$$V(G) = 16 - 9 + 2(1) = 16 - 9 + 2 = 9 + 2 = 11$$