|  |
| --- |
| Prueba de Caja Blanca |
| ***“Proyecto tienda de ropa”*** |
| Sprint 4  V4.1 |
| **GRUPO N#2**  **Integrantes:**  Gabriel Del Salto Kevin Lechón Paúl Jaramillo |

# Fecha: 31/07/2023

**Prueba Caja Blanca 1**

**Registro de cuentas de clientes**

**Código fuente:**

**{%- if form.errors -%}**

**<h2 class="form\_\_message" tabindex="-1" autofocus>**

**<svg aria-hidden="true" focusable="false">**

**<use href="#icon-error" />**

**</svg>**

**{{ 'templates.contact.form.error\_heading' | t }}**

**</h2>**

**<ul>**

**{%- for field in form.errors -%}**

**<li>**

**{%- if field == 'form' -%}**

**{{ form.errors.messages[field] }}**

**{%- else -%}**

**<a href="#RegisterForm-{{ field }}">**

**{{ form.errors.translated\_fields[field] | capitalize }}**

**{{ form.errors.messages[field] }}**

**</a>**

**{%- endif -%}**

**</li>**

**{%- endfor -%}**

**</ul>**

**{%- endif -%}**

**{%- if form.first\_name %}**

**value="{{ form.first\_name }}"**

**{%- endif -%}**

**{%- if form.last\_name %}**

**value="{{ form.last\_name }}"**

**{%- endif -%}**

**{%- if form.email %}**

**value="{{ form.email }}"**

**{%- endif -%}**

**{% if form.errors contains 'email' %}**

**aria-invalid="true"**

**aria-describedby="RegisterForm-email-error"**

**{%- endif -%}**

**{%- if form.errors contains 'email' -%}**

**<span id="RegisterForm-email-error" class="form\_\_message">**

**<svg aria-hidden="true" focusable="false">**

**<use href="#icon-error" />**

**</svg>**

**{{ form.errors.translated\_fields.email | capitalize }}**

**{{ form.errors.messages.email }}.**

**</span>**

**{%- endif -%}**

**{% if form.errors contains 'password' %}**

**aria-invalid="true"**

**aria-describedby="RegisterForm-password-error"**

**{%- endif -%}**

**{%- if form.errors contains 'password' -%}**

**<span id="RegisterForm-password-error" class="form\_\_message">**

**<svg aria-hidden="true" focusable="false">**

**<use href="#icon-error" />**

**</svg>**

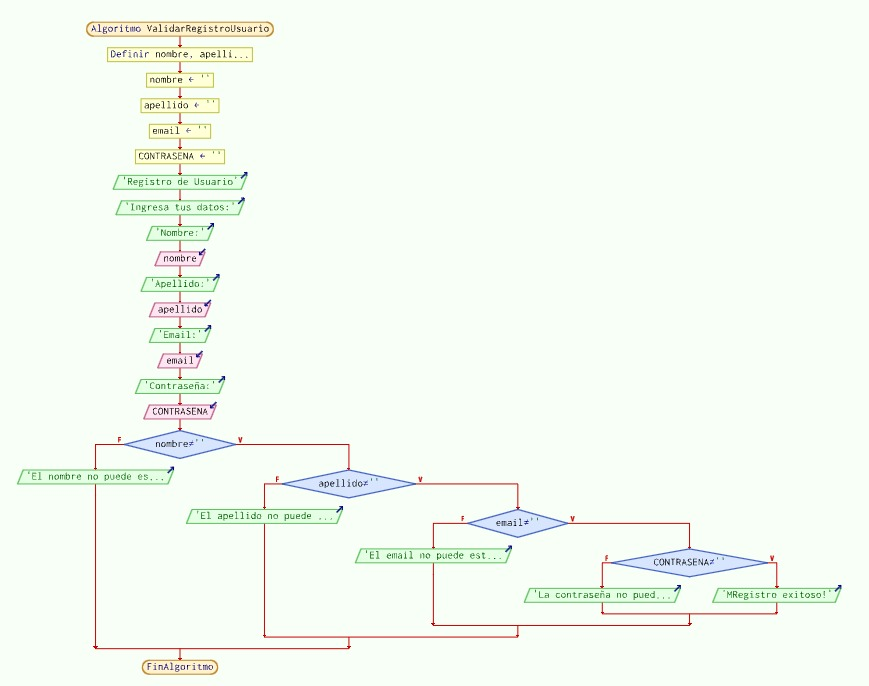
**{{ form.errors.translated\_fields.password | capitalize }}**

**{{ form.errors.messages.password }}.**

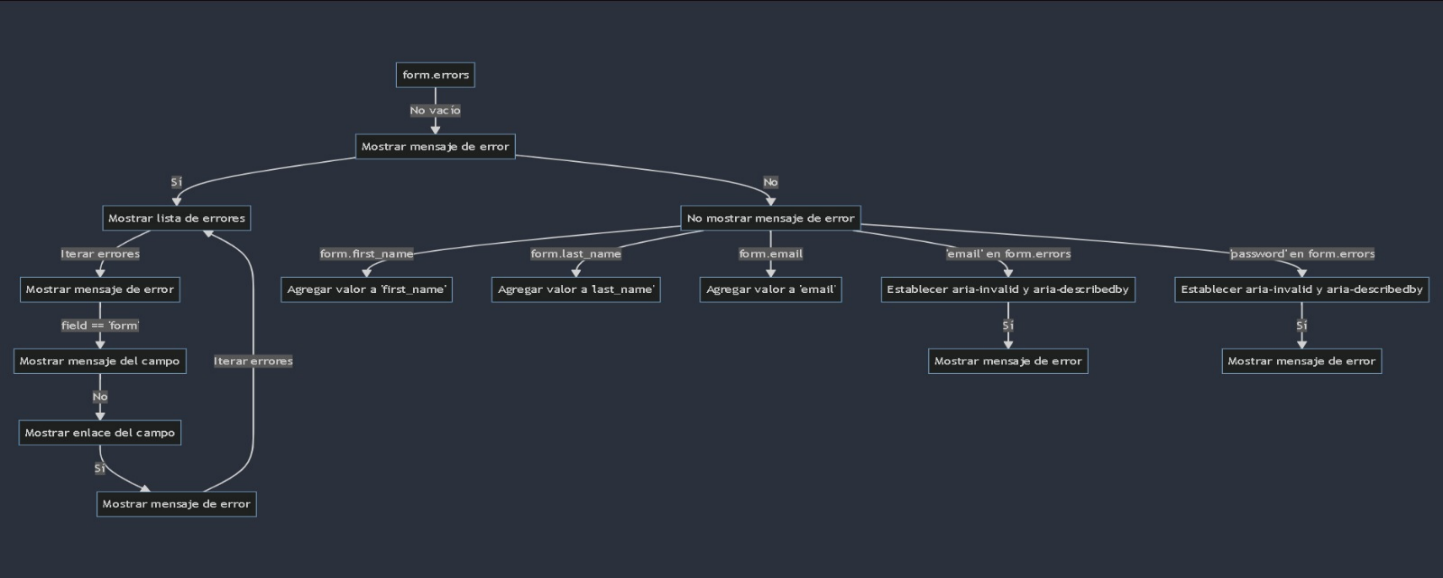
**</span>**

**{%- endif -%}**

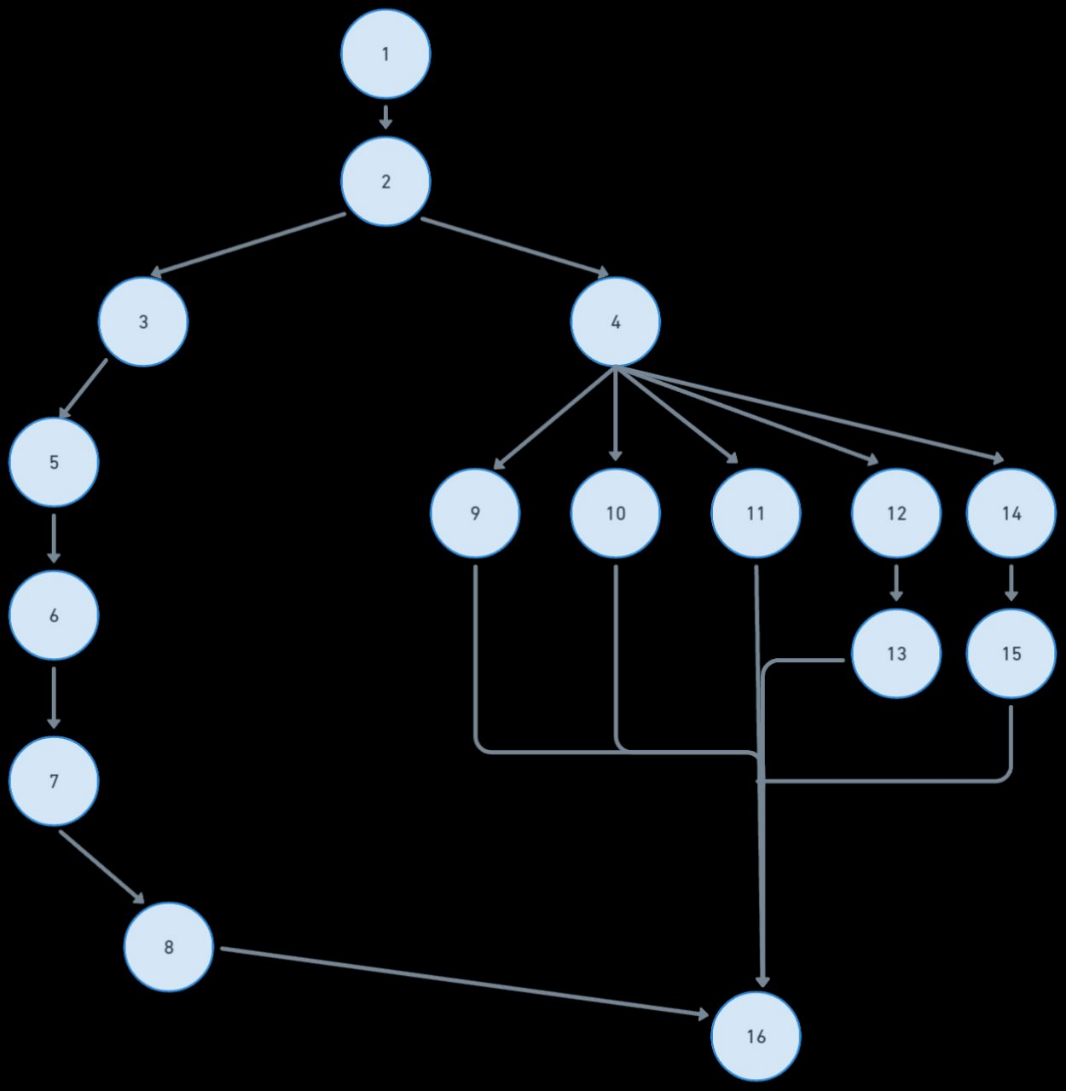
**Diagrama UML algoritmo:**

****

**Diagrama UML:**

****

**Grafo:**

****

**RUTAS:**

R1: 1,2,3,5,6,7,8,16

R2: 1,2,4,9,16

R3: 1,2,4,10,16

R4: 1,2,4,11,16

R5: 1,2,4,12,13,16

R6: 1,2,4,14,15,16

**Complejidad Ciclomática**

E: Número de aristas

N: Número de nodos

P: Número de nodos predicado

Aristas: 20

Nodos: 16

Predicados: 2

V(G)= Aristas - Nodos+2

V(G)=22-16+2

V(G)=8

**Requisito 3: Catálogo**

**Código fuente:**

<!DOCTYPE html>

<html>

<head>

<title>Formulario de Catálogo</title>

<script>

function validarFormulario() {

// Obtener los valores ingresados ​​en el formulario

var nombreProducto = document.getElementById("nombreProducto").value;

var precio = document.getElementById("precio").value;

var cantidad = document.getElementById("cantidad").value;

var tallas = document.getElementById("tallas").value;

var color = document.getElementById("color").value;

// Validar que los campos no estén vacíos

if (nombreProducto === "") {

alert("Por favor, ingresa el nombre del producto.");

return false;

}

if (precio === "") {

alert("Por favor, ingresa el precio del producto.");

return false;

}

// Validar que el precio no sea negativo

if (parseFloat(precio) < 0) {

alert("El precio no puede ser un número negativo.");

return false;

}

// Si todos los campos están completos y la validación es exitosa

// puedes realizar alguna acción adicional aquí, como enviar el formulario a un servidor

// Por ahora, simplemente mostramos una alerta para demostrar que el formulario es válido

alert("¡El formulario es válido!");

// Limpiar los campos del formulario

document.getElementById("nombreProducto").value = "";

document.getElementById("precio").value = "";

document.getElementById("cantidad").value = "";

document.getElementById("tallas").value = "";

document.getElementById("color").value = "";

return true;

}

</script>

</head>

<body>

<h1>Formulario de Catálogo</h1>

<form onsubmit="return validarFormulario();">

<label for="nombreProducto">Nombre del producto:</label>

<input type="text" id="nombreProducto" name="nombreProducto" required><br>

<label for="precio">Precio:</label>

<input type="number" id="precio" name="precio" min="0" required><br>

<label for="cantidad">Cantidad:</label>

<input type="number" id="cantidad" name="cantidad" min="0"><br>

<label for="tallas">Tallas:</label>

<input type="text" id="tallas" name="tallas"><br>

<label for="color">Color:</label>

<input type="text" id="color" name="color"><br>

<input type="submit" value="Enviar">

</form>

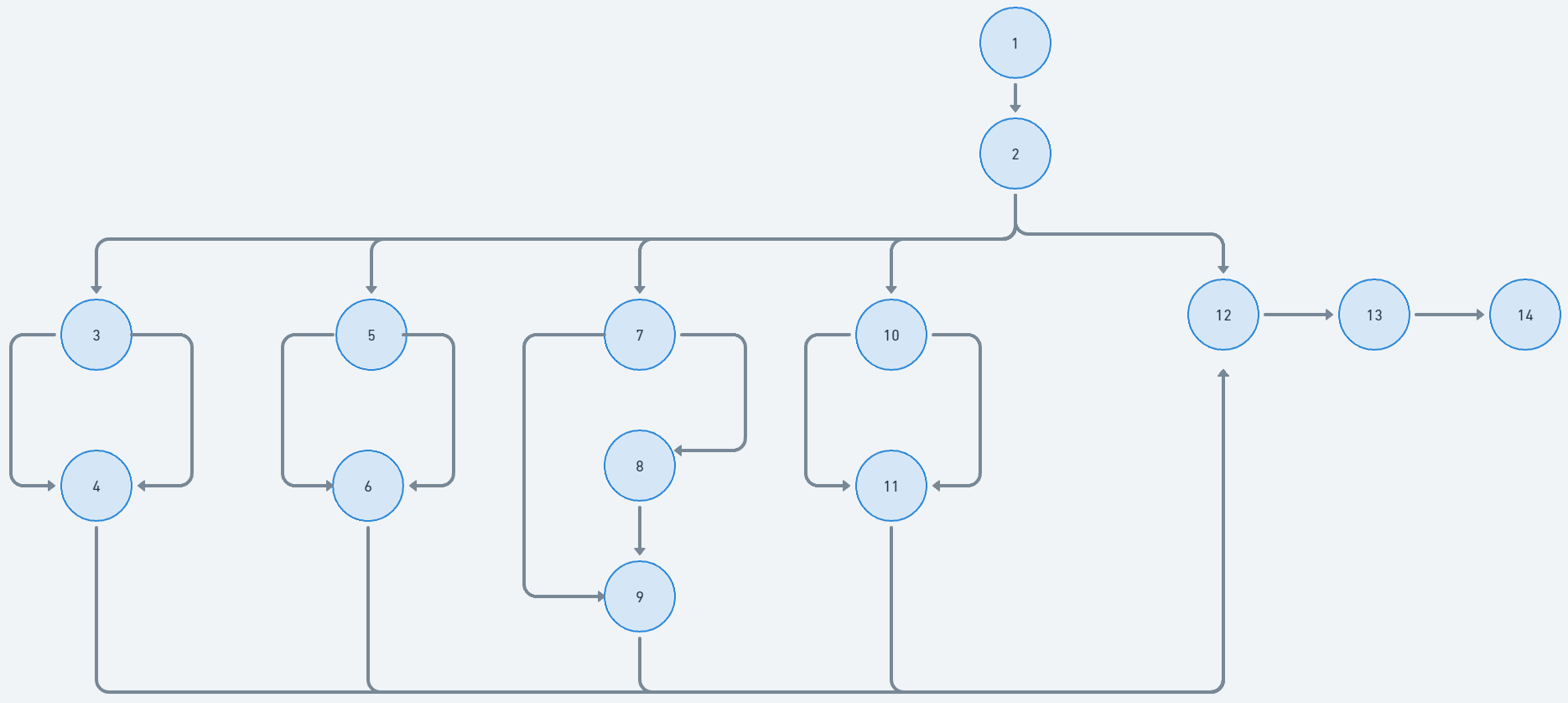
</body>

</html>

# Diagrama UML:

# 

**Grafo:**



**Rutas: (6 rutas)**

R1: 1,2,3,4,12,13,14

R2: 1,2,5,6,12,13,14

R3: 1,2,7,9,12,13,14

R4: 1,2,7,8,9,12,13,14

R5: 1,2,10,11,12,13,14

R6: 1,2,12,13,14

**Definición:**

Cuantas Aristas: 21

Cuantos Nodos: 14

Cuantos nodos Predicados: 5

# Complejidad Ciclo matica:

**V(G)=A-n+2 V(G)=**18-14+2=6

Conclusión: En conclusión, los resultados indican que la complejidad ciclomática de este programa es de 6. Esto significa que existen 6 caminos independientes en la estructura del programa, lo cual puede ser útil para evaluar su complejidad y realizar análisis de cobertura de código.

**Requisito 4: Carrito de compras**

<td class="cart-item\_\_details">

{%- if settings.show\_vendor -%}

<p class="caption-with-letter-spacing">{{ item.product.vendor }}</p>

{%- endif -%}

<a href="{{ item.url }}" class="cart-item\_\_name h4 break">{{ item.product.title | escape }}</a>

{%- if item.original\_price != item.final\_price -%}

<div class="cart-item\_\_discounted-prices">

<span class="visually-hidden">

{{ 'products.product.price.regular\_price' | t }}

</span>

<s class="cart-item\_\_old-price product-option">

{{- item.original\_price | money -}}

</s>

<span class="visually-hidden">

{{ 'products.product.price.sale\_price' | t }}

</span>

<strong class="cart-item\_\_final-price product-option">

{{ item.final\_price | money }}

</strong>

</div>

{%- else -%}

<div class="product-option">

{{ item.original\_price | money }}

</div>

{%- endif -%}

{%- if item.product.has\_only\_default\_variant == false

or item.properties.size != 0

or item.selling\_plan\_allocation != null

-%}

<dl>

{%- if item.product.has\_only\_default\_variant == false -%}

{%- for option in item.options\_with\_values -%}

<div class="product-option">

<dt>{{ option.name }}:</dt>

<dd>{{ option.value }}</dd>

</div>

{%- endfor -%}

{%- endif -%}

{%- for property in item.properties -%}

{%- assign property\_first\_char = property.first | slice: 0 -%}

{%- if property.last != blank and property\_first\_char != '\_' -%}

<div class="product-option">

<dt>{{ property.first }}:</dt>

<dd>

{%- if property.last contains '/uploads/' -%}

<a href="{{ property.last }}" class="link" target="\_blank">

{{ property.last | split: '/' | last }}

</a>

{%- else -%}

{{ property.last }}

{%- endif -%}

</dd>

</div>

{%- endif -%}

{%- endfor -%}

</dl>

<p class="product-option">{{ item.selling\_plan\_allocation.selling\_plan.name }}</p>

{%- endif -%}

<ul class="discounts list-unstyled" role="list" aria-label="{{ 'customer.order.discount' | t }}">

{%- for discount in item.discounts -%}

<li class="discounts\_\_discount">

{%- render 'icon-discount' -%}

{{ discount.title }}

</li>

{%- endfor -%}

</ul>

</td>

<td class="cart-item\_\_totals right medium-hide large-up-hide">

<div class="loading-overlay hidden">

<div class="loading-overlay\_\_spinner">

<svg

aria-hidden="true"

focusable="false"

class="spinner"

viewBox="0 0 66 66"

xmlns="http://www.w3.org/2000/svg"

>

<circle class="path" fill="none" stroke-width="6" cx="33" cy="33" r="30"></circle>

</svg>

</div>

</div>

<div class="cart-item\_\_price-wrapper">

{%- if item.original\_line\_price != item.final\_line\_price -%}

<dl class="cart-item\_\_discounted-prices">

<dt class="visually-hidden">

{{ 'products.product.price.regular\_price' | t }}

</dt>

<dd>

<s class="cart-item\_\_old-price price price--end">

{{ item.original\_line\_price | money }}

</s>

</dd>

<dt class="visually-hidden">

{{ 'products.product.price.sale\_price' | t }}

</dt>

<dd class="price price--end">

{{ item.final\_line\_price | money }}

</dd>

</dl>

{%- else -%}

<span class="price price--end">

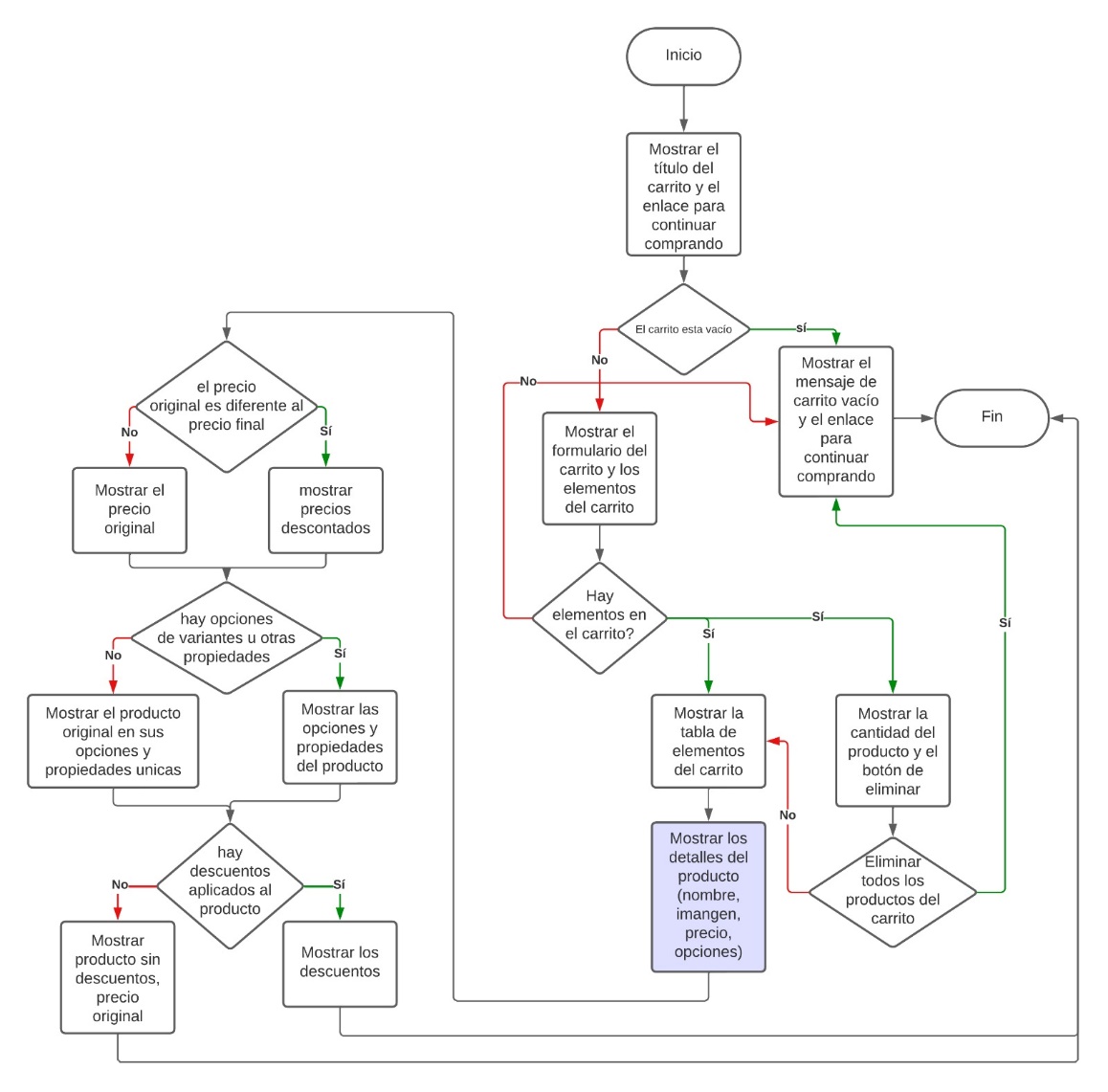
{{ item.original\_line\_price | money }}

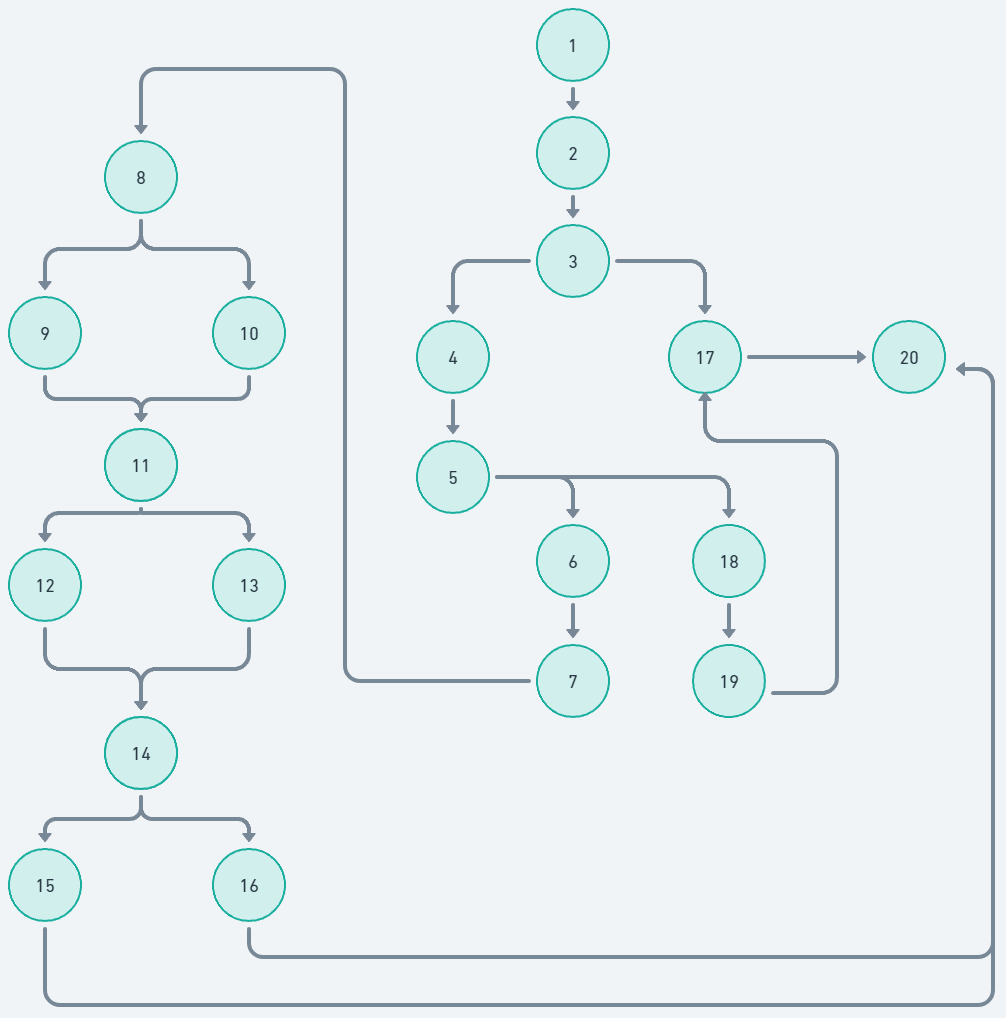
</span>

{%- endif -%}

{% endschema %}

**Diagrama UML**





**Rutas: (6 rutas)**

R1: 1,2,3,17,20

R2: 1,2,3,4,5,18,19,20

R3: 1,2,3,4,5,6,7,8,9,11,12,14,15,20

R4: 1,2,3,4,5,6,7,8,10,11,13,14,16,20

R5: 1,2,3,4,5,6,7,8,9,11,13,14,15,20

R6: 1,2,3,4,5,6,7,8,10,11,12,14,16,20

**Definición:**

Cuantas Aristas: 24

Cuantos Nodos: 20

Cuantos nodos Predicados: 5

# Complejidad Ciclo matica:

**V(G)=A-n+2 V(G)=**24-20+2=6

**Requisito 5: Botón de Contacto**

**Código:**

<!DOCTYPE html>

<html>

<head>

<title>Contacto</title>

</head>

<body>

<h1>Contacto</h1>

<form id="contactForm">

<label for="nombre">Nombre:</label>

<input type="text" id="nombre" name="nombre" required>

<br>

<label for="apellido">Apellido:</label>

<input type="text" id="apellido" name="apellido" required>

<br>

<label for="correo">Correo electrónico:</label>

<input type="email" id="correo" name="correo" required>

<br>

<label for="telefono">Número de teléfono:</label>

<input type="text" id="telefono" name="telefono" required>

<br>

<label for="comentario">Comentario:</label>

<textarea id="comentario" name="comentario" required></textarea>

<br>

<input type="submit" value="Enviar">

</form>

<script>

document.getElementById("contactForm").addEventListener("submit", function (event) {

event.preventDefault();

if (validateForm()) {

// Aquí debes implementar el código para enviar el formulario

// Puedes utilizar una API de correo para enviar los datos al dueño de la página

// Por ejemplo, puedes hacer una petición HTTP POST utilizando Fetch o Axios hacia tu servidor con los datos del formulario

// La implementación del envío de correo depende del lenguaje de backend que estés utilizando en tu servidor

alert("Formulario enviado correctamente.");

}

});

function validateForm() {

var telefono = document.getElementById("telefono").value;

var correo = document.getElementById("correo").value;

// Validar que el número de teléfono solo contenga números

if (!/^\d+$/.test(telefono)) {

alert("El número de teléfono debe contener solo números.");

return false;

}

// Validar que el campo de correo no esté vacío y sea un correo válido

if (!correo) {

alert("Debes ingresar un correo electrónico.");

return false;

} else if (!isValidEmail(correo)) {

alert("Ingresa un correo electrónico válido.");

return false;

}

return true;

}

function isValidEmail(email) {

// Expresión regular para validar el formato del correo electrónico

var emailRegex = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;

return emailRegex.test(email);

}

</script>

</body>

</html>

<!DOCTYPE html>

<html>

<head>

<title>Formulario de Pago</title>

<style>

body {

font-family: Arial, sans-serif;

padding: 20px;

}

label {

display: block;

margin-bottom: 5px;

}

input[type="text"], input[type="number"] {

width: 200px;

padding: 5px;

margin-bottom: 10px;

}

select {

width: 205px;

padding: 5px;

margin-bottom: 10px;

}

button {

padding: 10px 20px;

background-color: #007bff;

color: #fff;

border: none;

cursor: pointer;

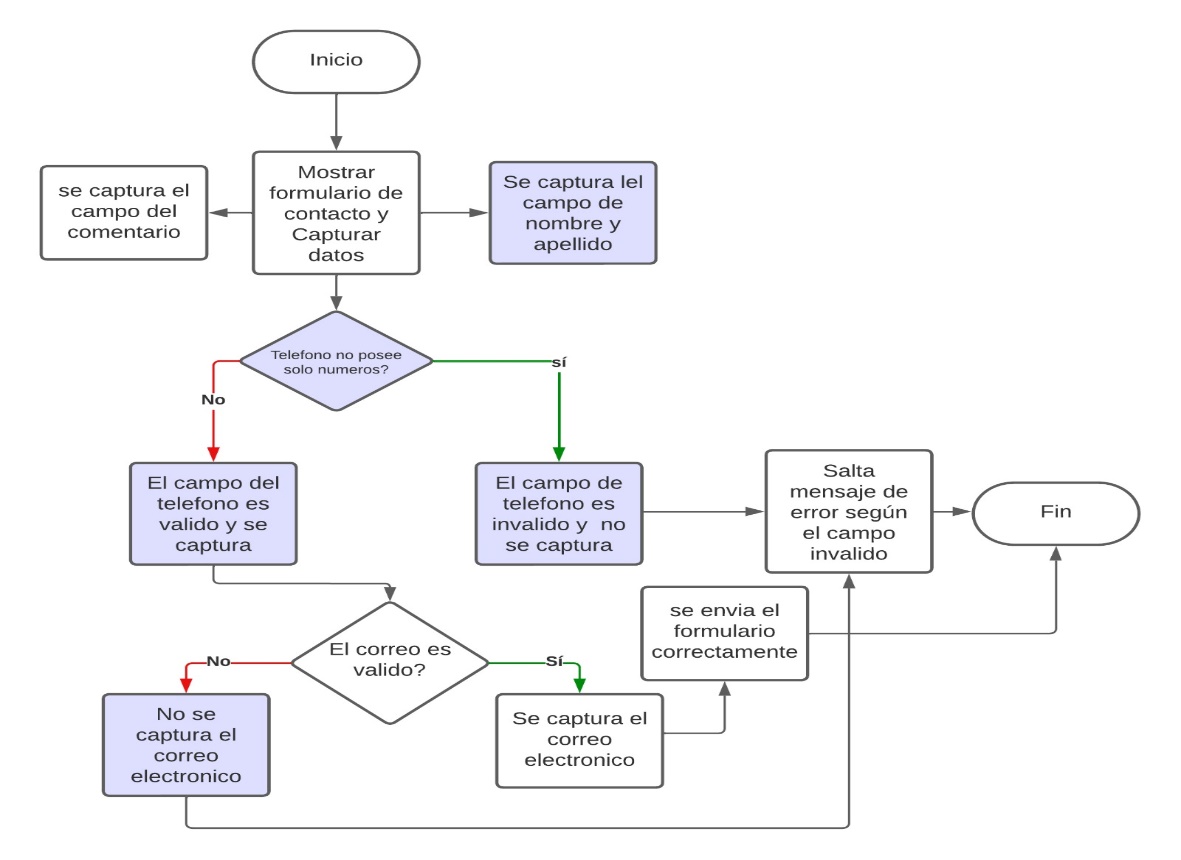
}

button:hover {

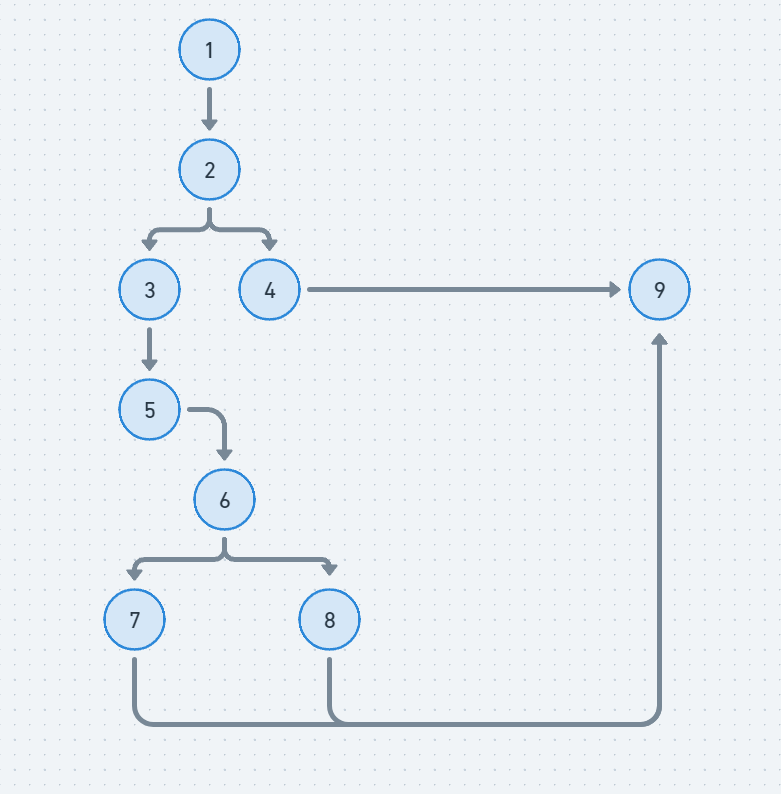
background-color: #0056b3;

}

**Diagrama UML**



**Grafo**

****

**Rutas: (3 rutas)**

R1: 1,2,4,9

R2: 1,2,3,5,6,7,9

R3: 1,2,3,5,6,8,9

**Definición:**

Cuantas Aristas: 10

Cuantos Nodos: 9

Cuantos nodos Predicados: 2

# Complejidad Ciclo matica:

**V(G)=A-n+2 V(G)=**10-9+2=3

**Requisito 6: Método de pago**

**Código:**

</style>

</head>

<body>

<h1>Información de Pago</h1>

<form>

<label for="nombre-titular">Nombre del Titular:</label>

<input type="text" id="nombre-titular" required>

<label for="numero-tarjeta">Número de Tarjeta:</label>

<input type="number" id="numero-tarjeta" required>

<label for="mes-expiracion">Mes de Expiración:</label>

<select id="mes-expiracion" required>

<option value="" disabled selected>Seleccione un mes</option>

<option value="01">Enero</option>

<option value="02">Febrero</option>

<!-- Agregar más opciones de meses -->

</select>

<label for="anio-expiracion">Año de Expiración:</label>

<select id="anio-expiracion" required>

<option value="" disabled selected>Seleccione un año</option>

<option value="2023">2023</option>

<option value="2024">2024</option>

<!-- Agregar más opciones de años -->

</select>

<label for="cvv">CVV:</label>

<input type="number" id="cvv" required>

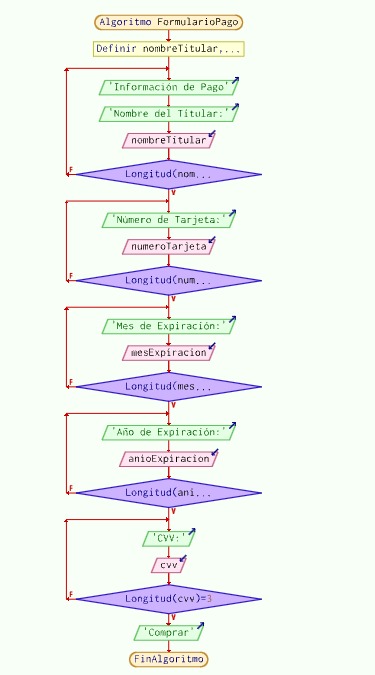
<button type="submit">Comprar</button>

</form>

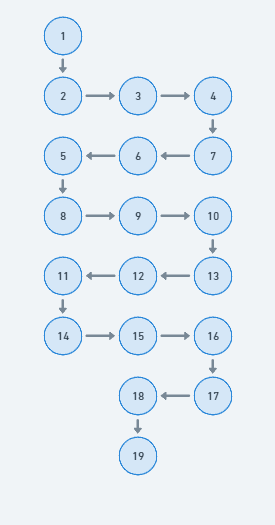
</body>

</html>

**Diagrama UML**



Grafo:



**Rutas: (1 ruta)**

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

**Definición:**

Cuantas Aristas: 18

Cuantos Nodos: 19

Cuantos nodos Predicados: 0

# Complejidad Ciclo matica:

**V(G)=A-n+2 V(G)=**18-19+2=1

**Requisito 7: Botón de Redes Sociales**

<!DOCTYPE html>

<html>

<head>

<title>Red Social</title>

<link rel="stylesheet" type="text/css" href="styles.css">

</head>

<body>

<h1>Red Social</h1>

<p>¿Te gustaría visitar la página web? (Sí o No)</p>

<button class="si-button">Sí</button>

<button class="no-button">No</button>

<div class="enlace-instagram">

<p>Enlace de Instagram.</p>

<p>Registrar cuenta</p>

</div>

<div class="gracias-mensaje">

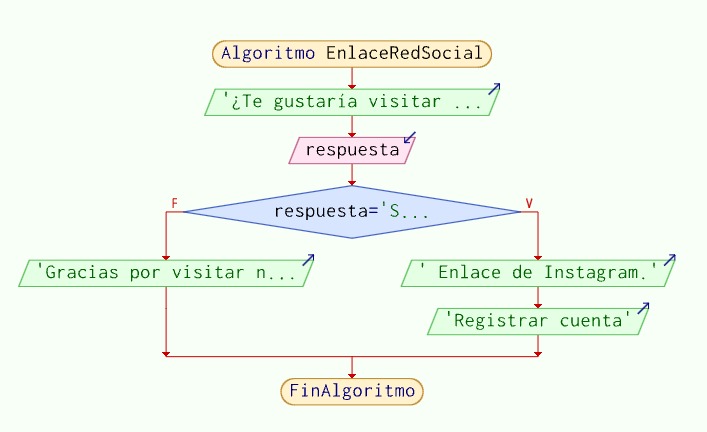
<p>Gracias por visitar nuestra red social.</p>

</div>

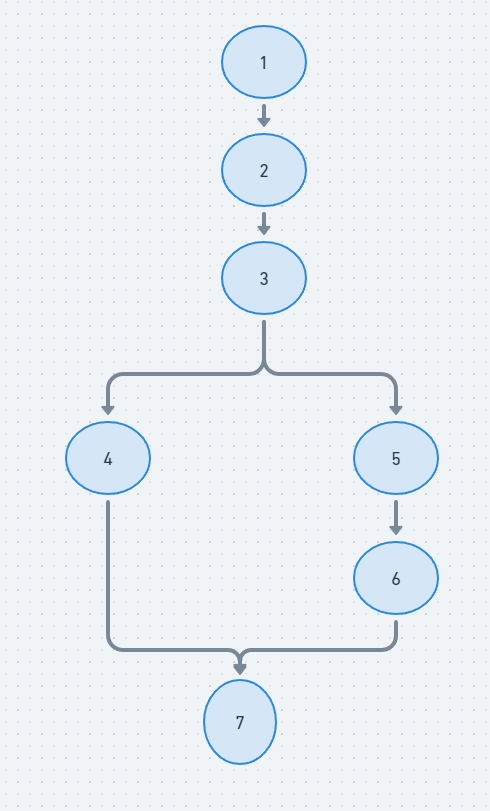
</body>

</html>

**Diagrama UML**

****

**Grafo:**



**Rutas: (2 ruta)**

R1: 1,2,3,4,7

R2: 1,2,3,5,6,7

**Definición:**

Cuantas Aristas: 7

Cuantos Nodos: 7

Cuantos nodos Predicados: 1

# Complejidad Ciclo matica:

**V(G)=A-n+2 V(G)=**7-7+2=2