

DOCUMENTACION DEL SERVICIO Comprar.jsp



Nicolás Santiago Hernández Arias

Servicio Nacional de Aprendizaje
SENA

Centro de la Tecnología del Diseño y la Productividad Empresarial

Análisis y Desarrollo de Software
2758348

Milton Iván Barbosa Gaona

Facatativá - Cundinamarca
2024

DESCRIPCIÓN DEL SERVICIO

El servicio "**Comprar**" corresponde a la página JSP encargada de mostrar el detalle del producto seleccionado para completar la compra. Esta página recibe un parámetro con el **ID del producto** y muestra la información relevante, como su imagen, descripción y precio. Si el producto no es encontrado, informa al usuario con un mensaje adecuado. También proporciona opciones para finalizar la compra o cancelar la operación.

PROPÓSITO

El propósito de este servicio es proporcionar al usuario una interfaz donde pueda confirmar la compra del producto seleccionado o cancelar la operación. Este componente es fundamental en la experiencia de compra, ya que permite al usuario revisar los detalles del producto antes de proceder con el pago.

Método HTTP Utilizado

- **GET:** Muestra los detalles del producto en función del parámetro `productoId` recibido en la solicitud.

Parámetros de Entrada

- **productoId (String):** ID del producto seleccionado para la compra. Se pasa como un parámetro en la URL:
`/comprar.jsp?productoId=1`

Respuesta del Servicio

- **Código 200 (OK):** La página se carga correctamente con los detalles del producto, si se encuentra.
- **Mensaje "Producto no encontrado":** Si el producto no está disponible, muestra un mensaje adecuado al usuario.

CODIGO FUENTE

```
<% @page contentType="text/html" pageEncoding="UTF-8"%>

<% @page import="java.util.HashMap"%>

<!DOCTYPE html>

<html lang="es">
```

```
<head>
```

```
<meta charset="UTF-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
```

```
<title>Beta Moda - Comprar</title>
```

```
<style>
```

```
/* Tus estilos previos */
```

```
.header {
```

```
    background-color: #4CAF50;
```

```
    color: white;
```

```
    padding: 15px;
```

```
    text-align: center;
```

```
}
```

```
.container {
```

```
    display: flex;
```

```
    flex-direction: column;
```

```
    align-items: center;
```

```
    padding: 20px;
```

```
}
```

```
.product {
```

```
    display: flex;
```

```
    flex-direction: column;
```

```
    align-items: center;
```

```
background-color: white;

border: 1px solid #ccc;

border-radius: 10px;

padding: 20px;

box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

width: 300px;

margin-bottom: 20px;
}

.product img {

    max-width: 100%;

    border-radius: 10px;
}

.product-details h3 {

    margin: 10px 0;

    font-size: 18px;
}

.product-details p {

    color: #666;
}

.price {

    font-weight: bold;

    color: #4CAF50;
```

```
}  
  
.buttons button {  
    margin: 5px;  
    padding: 10px 20px;  
    border: none;  
    border-radius: 5px;  
    cursor: pointer;  
    color: white;  
}  
  
.buy {  
    background-color: #4CAF50;  
}  
  
.buy:hover {  
    background-color: #45a049;  
}  
  
.cancel {  
    background-color: #f44336;  
}  
  
.cancel:hover {  
    background-color: #d32f2f;  
}  
  
</style>
```

```
</head>
```

```
<body>
```

```
    <div class="header">
```

```
        <h1>Beta Moda</h1>
```

```
        <p>Completa tu compra</p>
```

```
    </div>
```

```
    <div class="container">
```

```
        <h2>Carrito de Compras</h2>
```

```
        <%
```

```
            // Definición de los productos disponibles
```

```
            HashMap<String, String[]> productos = new HashMap<>();
```

```
            productos.put("1", new String[]{"Producto 1", "Descripción del producto 1",  
"$50,000 COP", "imagenes/Mercurial.png"});
```

```
            productos.put("2", new String[]{"Producto 2", "Descripción del producto 2",  
"$70,000 COP", "imagenes/traje.png"});
```

```
            // Obtención del ID del producto desde los parámetros de la solicitud
```

```
            String productoId = request.getParameter("productoId");
```

```
            String[] producto = productos.get(productoId);
```

```
            if (producto != null) {
```


%>

<div class="product">

">

<div class="product-details">

<h3><%= producto[0] %></h3>

<p><%= producto[1] %></p>

<p class="price"><%= producto[2] %></p>

</div>

</div>

<div class="buttons">

<button class="buy">Comprar</button>

<button

class="cancel"

onclick="window.location.href='index.jsp'">Cancelar</button>

</div>

<%

} else {

%>

<p>Producto no encontrado.</p>

<button

class="cancel"

onclick="window.location.href='index.jsp'">Regresar</button>

<%

}

```
%>

</div>

</body>

</html>
```

1. Explicación del Código

2. Cabecera JSP (<%@page contentType="text/html" pageEncoding="UTF-8"%>):

Indica que el contenido de la página es HTML y utiliza la codificación UTF-8 para soportar caracteres especiales.

3. Importación de Java (<%@page import="java.util.HashMap"%>):

Importa la clase HashMap de Java, que se usa para almacenar los productos disponibles.

4. Definición de Productos:

- Los productos disponibles se guardan en un HashMap con su ID como clave.
- Cada producto contiene un nombre, descripción, precio y la URL de su imagen.

5. Obtención del Producto Seleccionado:

- Se obtiene el ID del producto desde los parámetros de la solicitud (request.getParameter("productoId")).
- Si el producto existe en el HashMap, se muestra su información en la página.
- Si no se encuentra, muestra un mensaje de "Producto no encontrado".

6. HTML y CSS:

- Se utiliza CSS para dar estilo a la página, incluyendo la cabecera y las tarjetas de producto.
- Los botones proporcionan opciones para comprar o cancelar la operación.

7. Interacción:

- Si el usuario hace clic en Cancelar, se redirige a index.jsp.
- El botón Comprar actualmente no tiene funcionalidad adicional, pero puede integrarse con un servicio de pago en el futuro.

Manejo de Errores

- **Producto no Encontrado:**

Si el ID del producto no existe en el HashMap, la página muestra un mensaje indicando que el producto no fue encontrado.

- **Cancelación:**

Si el usuario decide cancelar, es redirigido a la página principal (index.jsp).

Dependencias

- **index.jsp:** La página depende de index.jsp para la redirección al cancelar la compra.
- **Imagenes:** La página requiere que las imágenes de los productos estén disponibles en la carpeta imagenes/.