

Barra de Navegación con Bootstrap



Modificación del `app.component.html` para agregar una barra de navegación con Bootstrap

En esta guía, explicaremos cómo se modificó el archivo `app.component.html` para agregar una barra de navegación con Bootstrap, así como la simplificación del archivo `app.component.ts` y la eliminación del archivo CSS asociado.

Paso 1: Agregar una barra de navegación con Bootstrap en `app.component.html`

Código `app.component.html`

```
<div class="container">
  <nav class="navbar navbar-expand-lg navbar-dark" style="background-color: #8c0aef;">
    <div class="container-fluid">
      <a class="navbar-brand" href="/">Sistema de Inventarios</a>
      <button class="navbar-toggler" type="button" data-bs-toggle="collapse"
        data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false"
        aria-label="Toggle navigation">
```

```
<span class="navbar-toggler-icon"></span>
</button>
<div class="collapse navbar-collapse" id="navbarNav">
  <ul class="navbar-nav">
    <li class="nav-item">
      <a class="nav-link active" aria-current="page" href="/">Inicio</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" href="/agregar-producto">Agregar Producto</a>
    </li>
  </ul>
</div>
</div>
</nav>
</div>

<app-producto-lista/>
```

Paso 2: Explicación de los elementos y clases de Bootstrap utilizados

1. **Contenedor principal (`div.container`)**
 - Envuelve la barra de navegación para que tenga márgenes adecuados y no ocupe toda la pantalla.
2. **Barra de navegación (`nav.navbar`)**
 - Se usa la clase `navbar` para definir una barra de navegación.
 - `navbar-expand-lg`: Permite que la barra sea expandible en pantallas grandes.
 - `navbar-dark`: Define que el texto de la barra sea blanco.
 - `style="background-color: #8c0aeef;"`: Se usa un color de fondo personalizado.
3. **Contenedor fluido (`div.container-fluid`)**
 - Permite que la barra de navegación ocupe todo el ancho disponible.
4. **Logotipo o nombre del sistema (`a.navbar-brand`)**
 - Usa `href`, para agregar el título de la aplicación en el menú de navegación.
5. **Botón de colapso (`button.navbar-toggler`)**
 - Es un botón que aparece en pantallas pequeñas para expandir o colapsar la barra de navegación.
 - `data-bs-toggle="collapse"` y `data-bs-target="#navbarNav"` indican que colapsará el menú con el ID `navbarNav`.
6. **Menú de navegación (`div.collapse.navbar-collapse`)**
 - Contiene los elementos del menú y se colapsa en pantallas pequeñas.
7. **Lista de elementos (`ul.navbar-nav`)**
 - `li.nav-item`: Define cada opción del menú.

- o `a.nav-link`: Se usa para los enlaces dentro del menú.
-

Paso 3: Simplificación del `app.component.ts`

El archivo `app.component.ts` se simplificó para eliminar configuraciones innecesarias y hacer uso de standalone components, que ya es la configuración por default en la última versión de angular.

Código `app.component.ts`

```
import { Component } from '@angular/core';
import { ProductoListaComponent } from './producto-lista/producto-lista.component';

@Component({
  selector: 'app-root',
  imports: [ProductoListaComponent],
  templateUrl: './app.component.html'
})
export class AppComponent {
```

Explicación de los cambios:

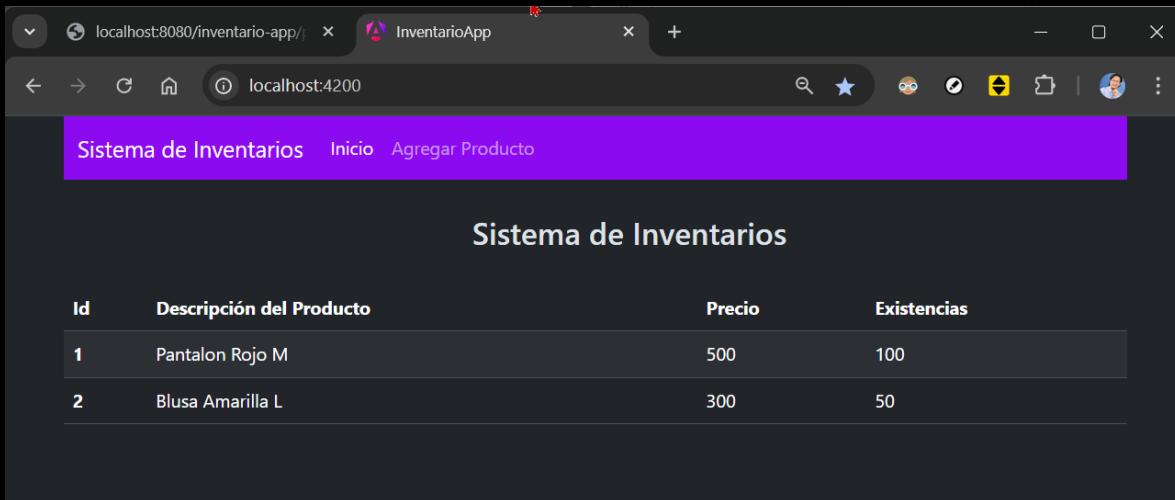
1. **Importación directa de `ProductoListaComponent` en `imports`**
 - o Se eliminó la necesidad de importar este componente desde un módulo, ya que es un componente standalone.
 2. **Simplificación del código**
 - o Se eliminó cualquier código innecesario, manteniendo solo la plantilla y la configuración mínima.
-

Paso 4: Eliminación del archivo CSS asociado

Dado que el color de la barra de navegación utiliza estilos CSS de Bootstrap, se eliminó el archivo CSS de `app.component.css`. Esto ayuda a simplificar la estructura del proyecto, evitando estilos dispersos en múltiples archivos.

Ejecutamos la aplicación

- Ejecutamos la aplicación con el comando: `ng serve -o`
- Entramos a la URL: <http://localhost:4200/>



Conclusión

- Se agregó una barra de navegación responsiva con Bootstrap en `app.component.html`.
- Se utilizó `routerLink` en lugar de `href` para navegación sin recargar la página.
- Se simplificó `app.component.ts` aprovechando standalone components.
- Se eliminó el archivo `app.component.css` ya que los estilos CSS son de Bootstrap.

Este enfoque mejora la estructura del código y mantiene el proyecto más limpio y eficiente en Angular.

Saludos!

Ing. Ubaldo Acosta

Fundador de [GlobalMentoring.com.mx](https://www.globalmentoring.com.mx)