

Agregar el Botón Editar con Angular



Creación del componente para editar un producto en Angular (Primera Parte)

En esta primera parte, crearemos el **componente para editar un producto**, agregaremos la ruta con el parámetro `id`, modificaremos la **tabla de productos** para incluir el **botón de editar** y agregaremos la **navegación al componente de edición**.

Paso 1: Crear el componente `EditarProductoComponent`

Ejecuta el siguiente comando en la terminal dentro del proyecto Angular:

```
ng g c editar-producto --skip-tests
```

Explicación del comando

- `ng g c editar-producto` → Genera un nuevo componente llamado `EditarProductoComponent`. `g` es de generate y `c` de component
- `--skip-tests` → Evita la creación del archivo de pruebas unitarias.

Archivos generados

<https://www.globalmentoring.com.mx>

```
/src/app/editar-producto/
└── editar-producto.component.ts
└── editar-producto.component.html
└── editar-producto.component.css (Se eliminará)
```

Paso 2: Eliminar el archivo CSS del componente

Como vamos a utilizar **Bootstrap** para los estilos, eliminamos el archivo innecesario.

1. Ubica el archivo:

```
/src/app/editar-producto/editar-producto.component.css
```

2. Elimínalo manualmente o con el siguiente comando:

```
rm src/app/editar-producto/editar-producto.component.css
```

Paso 3: Agregar la ruta para editar un producto en `app.routes.ts`

Código actualizado en `app.routes.ts`

```
import { Routes } from '@angular/router';
import { ProductoListaComponent } from './producto-lista/producto-lista.component';
import { AgregarProductoComponent } from './agregar-producto/agregar-producto.component';
import { EditarProductoComponent } from './editar-producto/editar-producto.component';

export const routes: Routes = [
  { path: 'productos', component: ProductoListaComponent },
  { path: '', redirectTo: 'productos', pathMatch: 'full' },
  { path: 'agregar-producto', component: AgregarProductoComponent },
  { path: 'editar-producto/:id', component: EditarProductoComponent }
];
```

Explicación de los cambios

- Se agregó la ruta `editar-producto/:id`:
 - `:id` es un **parámetro dinámico**, lo que permite acceder a un producto específico según su ID.
 - Cuando el usuario acceda a `http://localhost:4200/editar-producto/1`, Angular sabrá que debe cargar el componente `EditarProductoComponent` con el producto de ID 1.
-

Paso 4: Modificar la tabla para agregar el botón de editar en `producto-lista.component.html`

Código actualizado en `producto-lista.component.html`

```
<div class="container">
  <div class="container text-center" style="margin: 30px">
    <h3>Sistema de Inventarios</h3>
  </div>

  <table class="table table-striped table-hover align-middle">
    <thead class="table-dark">
      <tr>
        <th scope="col">Id</th>
        <th scope="col">Descripción del Producto</th>
        <th scope="col">Precio</th>
        <th scope="col">Existencias</th>
        <th scope="col" class="text-center">Acciones</th>
      </tr>
    </thead>
    <tbody>
      @for (producto of productos; track producto.idProducto) {
        <tr>
          <th scope="row">{{ producto.idProducto }}</th>
          <td>{{ producto.descripcion }}</td>
          <td>{{ producto.precio }}</td>
          <td>{{ producto.existencia }}</td>
          <td class="text-center">
            <div>
              <button (click)="editarProducto(producto.idProducto)"
                class="btn btn-warning btn-sm me-3">Editar</button>
            </div>
          </td>
        </tr>
      }
    </tbody>
  </table>
</div>
```

Explicación de los cambios

1. Se agregó una nueva columna en la tabla con el título "Acciones":

```
<th scope="col" class="text-center">Acciones</th>
```

- Esta columna será utilizada para los botones de edición.
2. Se agregó un botón de edición en cada fila de la tabla:

```
<button (click)="editarProducto(producto.idProducto)" class="btn btn-warning btn-sm me-3">Editar</button>
```

- **(click)="editarProducto(producto.idProducto)":** Llama a la función `editarProducto()` enviando el ID del producto.
- **class="btn btn-warning btn-sm me-3":**
 - `btn-warning`: Aplica un color amarillo (Bootstrap).
 - `btn-sm`: Reduce el tamaño del botón.
 - `me-3`: Agrega un margen derecho.

Paso 5: Agregar el método `editarProducto()` en `producto-lista.component.ts`

Código actualizado en `producto-lista.component.ts`

```
import { Component, inject } from '@angular/core';
import { Producto } from '../producto';
import { ProductoService } from '../producto.service';
import { Router } from '@angular/router';

@Component({
  selector: 'app-producto-lista',
  templateUrl: './producto-lista.component.html'
})
export class ProductoListaComponent {
  productos: Producto[] = [];

  // Nueva forma de inyectar servicios
  private productoServicio = inject(ProductoService);
  private enrutador = inject(Router);

  ngOnInit() {
    //Cargamos los productos
    this.obtenerProductos();
  }

  private obtenerProductos(): void {
    this.productoServicio.obtenerProductosLista().subscribe({
      next: (datos) => {
        this.productos = datos;
      }
    });
  }
}
```

```

    },
    error: (error) => {
      console.error("Error al obtener los productos", error);
    }
  );
}

editarProducto(id: number){
  this.enrutador.navigate(['editar-producto', id]);
}
}

```

Explicación de los cambios

1. Se inyectó `Router` para manejar la navegación:

```
private enrutador = inject(Router);
```

- Esto permite navegar a otras páginas dentro de la aplicación.

2. Se creó el método `editarProducto(id: number)`:

```
editarProducto(id: number) {
  this.enrutador.navigate(['editar-producto', id]);
}
```

- `this.enrutador.navigate(['editar-producto', id])`:
 - Cambia la URL a `http://localhost:4200/editar-producto/{id}` cuando se presiona el botón "Editar".
 - Esto carga el `EditarProductoComponent` con el ID correspondiente.

Paso 6: Ejecutar y probar la aplicación

1. Ejecutar la aplicación Angular:

```
ng serve -o
```

2. Abrir el navegador en <http://localhost:4200/>

Id	Descripción del Producto	Precio	Existencias	
1	Pantalon Rojo M	500	100	<button>Editar</button>
2	Blusa Amarilla L	300	50	<button>Editar</button>
3	Vaso Tequilero	200	30	<button>Editar</button>
4	Florero L	350	23	<button>Editar</button>

3. Verificar que en la tabla de productos aparezca el botón "Editar" en cada fila.
4. Hacer clic en el botón "Editar" de un producto:
 - o La URL del navegador cambiará a algo como:

<http://localhost:4200/editar-producto/1>

editar-producto works!

- o Esto indica que la funcionalidad de navegación trabaja correctamente.

Conclusión

- Se creó el componente `EditarProductoComponent` con `ng g c editar-producto --skip-tests`.
- Se eliminó el archivo CSS innecesario.
- Se agregó la ruta `editar-producto/:id` en `app.routes.ts`.
- Se modificó la tabla en `producto-lista.component.html` para agregar el botón "Editar".

- Se creó el método `editarProducto(id: number)` en `producto-lista.component.ts` para navegar a la edición.
- Al hacer clic en el botón "Editar", la URL cambia según el producto seleccionado.

Con estos cambios, la estructura para editar un producto ya está lista, y en la siguiente parte implementaremos el formulario de edición. 

Saludos!

Ing. Ubaldo Acosta

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