



Angular - Componentes

Profa. Maria Janaina da Silva Ferreira



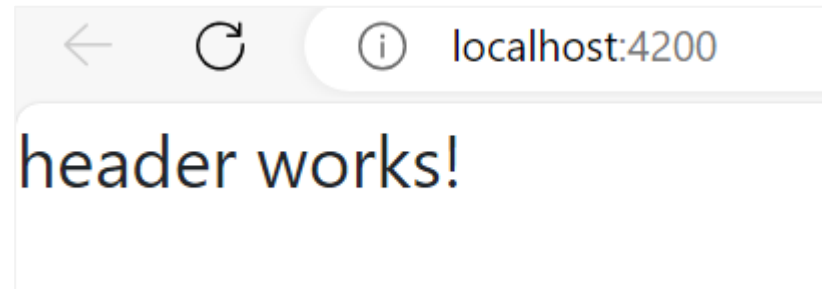
Configurações Adicionais

- Adicione a extensão 'Angular Extension Pack' (Loraine Groner) que vai facilitar o desenvolvimento
- Faça a instalação do framework Bootstrap
 - `npm install bootstrap@4.1.3 jquery@3.3.1 popper.js@1.14.3 --save`
- Adicione as referências no arquivo `angular.json`

```
27     "styles": [  
28       "node_modules/bootstrap/dist/css/bootstrap.css",  
29       "src/styles.css"  
30     ],  
31     "scripts": [  
32       "node_modules/jquery/dist/jquery.js",  
33       "node_modules/popper.js/dist/umd/popper.js",  
34       "node_modules/bootstrap/dist/js/bootstrap.js"  
35     ],
```

Configurações adicionais

- Execute novamente a aplicação:
 - `ng serve --open`
- Verifique se o Bootstrap foi aplicado no layout, ou seja, se a fonte do texto foi alterada.



Criando o componente header

- No seu componente de Header, vamos criar a área onde está o nome do seu SITE:

```
<div class="titulosite d-flex align-items-center">  
  <span>{{this.titulosite}}</span>  
</div>
```

HTML

```
titulosite = "Extensões FATEC"
```

TS

Criando o componente header

Para centralizar o texto na DIV. Usamos as classes:

`d-flex align-items-center`, características do Flexbox:

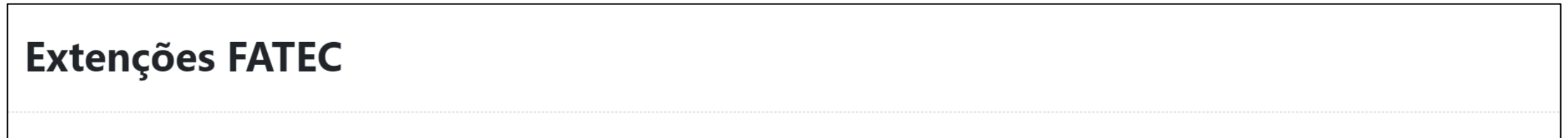
Foi adicionado o seguinte CSS:


TS

```
.titulosite{
  font-size: 40px;
  font-weight: bold;
  height: 112px;
  border-bottom: 1px dashed #d0d0d0;
  padding: 32px;
}
```

Criando o componente header

- Nosso site atualmente está assim:



- Vamos adicionar o ícone  no canto superior direito do seu site
- Para isso, inclua o código html do seu componente de Header:

```
Go to component
1 <div class="row">
2   <div class="titulosite d-flex align-items-center">
3     <span>{{this.titulosite}}</span>
4   </div>
5   <div class="titulosite d-flex align-items-center col-sm justify-content-end">
6     
7   </div>
8 </div>
```

Criando o componente header

- Agora vamos adicionar um menu flutuante que irá abrir ao clicar no seu ícone: 

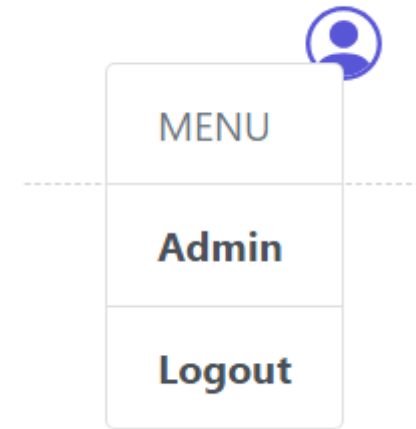
```
<div class="list-group menuusuario">  
  <li class="list-group-item disabled">MENU</li>  
  <a href="#" class="list-group-item list-group-item-action itemclicavel">Admin</a>  
  <a href="#" class="list-group-item list-group-item-action itemclicavel">Logout</a>  
</div>
```

Criando o componente header

- CSS:

```
8  .menuusuario{
9      position: absolute;
10     right: 32px;
11     top: 70px;
12     z-index: 1;
13 }
14 .itemclicavel{
15     font-weight: bold;
16 }
```

- Como ficou:



Criando o componente header

- Podemos agora condicionar o menu de opções ao clique no ícone 

Para isso vamos usar a diretiva ***ngIf** disponível no CommonModule do Angular.

Para usar essa diretiva, importe o CommonModule no header.component.ts:

```
2 | import { CommonModule } from '@angular/common';
3 |
4 | @Component({
5 |   selector: 'app-header',
6 |   standalone: true,
7 |   imports: [CommonModule],
8 |   templateUrl: './header.component.html',
9 |   styleUrls: ['./header.component.css']
10 | })
11 | export class HeaderComponent {
12 |   menuusuario = false;
```

Criando o componente header

- No html, adicione o evento de click a div do ícone:

```
<div class="titulosite d-flex align-items-center col-sm justify-content-end">
  <a (click)="this.menuusuario = !this.menuusuario">
    
  </a>
</div>
</div>
<div *ngIf="this.menuusuario" class="list-group menuusuario">
  <li class="list-group-item disabled">MENU</li>
  <a href="#" class="list-group-item list-group-item-action itemclicavel">Admin</a>
  <a href="#" class="list-group-item list-group-item-action itemclicavel">Logout</a>
</div>
```

- Adicione também a diretiva *ngif a div que exibe o menu

Criando o componente header

- No ts, crie a variável que será responsável por ocultar/exibir o menu

```
11 |  
12 | export class HeaderComponent {  
13 | |   menuusuario = false;  
14 | |   titulosite = "Extensões FATEC"  
15 | | }  
16 |
```

Criando o componente header

- Finalizamos o componente header. O código html ficou assim:

```
<> header.component.html M X
Go to component
1 <div class="row">
2   <div class="titulosite d-flex align-items-center">
3     <span>{{this.titulosite}}</span>
4   </div>
5   <div class="titulosite d-flex align-items-center col-sm justify-content-end">
6     <a (click)="this.menuusuario = !this.menuusuario" class="personcircle">
7       
8     </a>
9   </div>
10 </div>
11 <div *ngIf="this.menuusuario" class="list-group menuusuario">
12   <li class="list-group-item disabled">MENU</li>
13   <a href="#" class="list-group-item list-group-item-action itemclicavel">Admin</a>
14   <a href="#" class="list-group-item list-group-item-action itemclicavel">Logout</a>
15 </div>
```

Criando o componente header

- O Css ficou assim:

```
# header.component.css M x
1  .titulosite{
2      font-size: 40px;
3      font-weight: bold;
4      height: 112px;
5      border-bottom: 1px dashed #d0d0d0;
6      padding: 32px;
7  }
8  .menuusuario{
9      position: absolute;
10     right: 32px;
11     top: 70px;
12     z-index: 1;
13 }
14 .itemclicavel{
15     font-weight: bold;
16 }
17 .personcircle{
18     cursor: pointer;
19 }
20 .row {
21     margin-right: 0px;
22     margin-left: 0px;
23 }
24
```

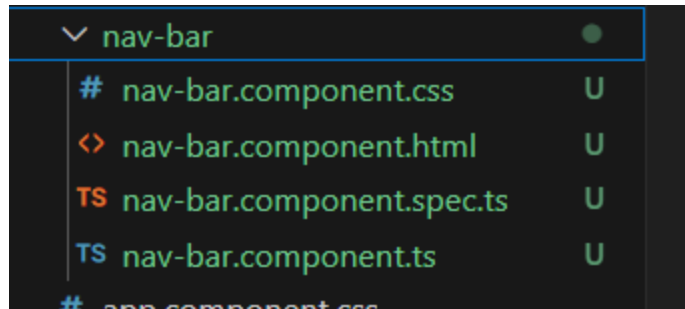
Criando o componente header

- E o ts:

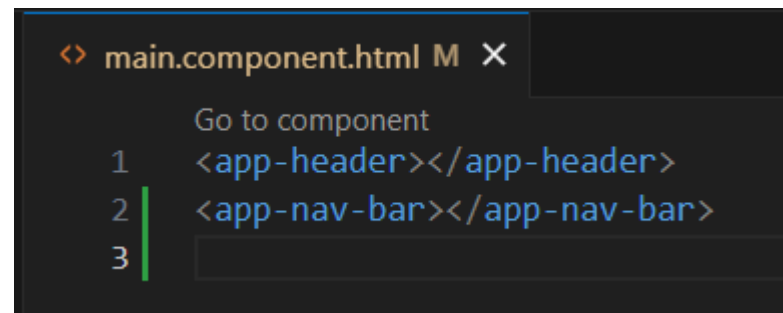
```
TS header.component.ts M X
1  import { Component } from '@angular/core';
2  import { CommonModule } from '@angular/common';
3
4  @Component({
5    selector: 'app-header',
6    standalone: true,
7    imports: [CommonModule],
8    templateUrl: './header.component.html',
9    styleUrls: ['./header.component.css']
10 })
11 export class HeaderComponent {
12   menuusuario = false;
13   titulosite = "Extensões FATEC"
14 }
```

Criando o componente navBar

- No terminal, digite o comando para criação do componente:
ng g component nav-bar



- Agora adicione o seu novo componente no seu arquivo main.html:



Criando o componente navBar

- Não esqueça de adicionar a referência ao componente nav-bar no seu arquivo main.ts

```
TS main.component.ts M X
1 import { Component } from '@angular/core';
2 import { HeaderComponent } from '../header/header.component';
3 import { NavBarComponent } from "../nav-bar/nav-bar.component";
4
5 @Component({
6   selector: 'app-main',
7   standalone: true,
8   imports: [HeaderComponent, NavBarComponent],
9   templateUrl: './main.component.html',
10  styleUrls: ['./main.component.css']
11 })
12 export class MainComponent {
13
14 }
```


Criando o componente nav-Bar

- No arquivo nav-bar.component.html, insira o navbar do Bootstrap usando o seguinte código:

```
nav-bar.component.html U X
Go to component
1 <div class="pos-f-t menuexpansivel">
2   <nav class="navbar navbar-light">
3     <button class="navbar-toggler"
4       type="button"
5       data-toggle="collapse"
6       data-target="#navbarToggleExternalContent"
7       aria-controls="navbarToggleExternalContent"
8       aria-expanded="false"
9       aria-label="Toggle navigation">
10      <span class="navbar-toggler-icon"></span>
11    </button>
12  </nav>
13  <div class="collapse" id="navbarToggleExternalContent">
14    <div class="list-group menuusuario bg-light p4">
15      <li class="list-group-item disabled">CATEGORIAS</li>
16      <a href="#" class="list-group-item list-group-item-action itemclicavel">Tecnologia da Informação</a>
17      <a href="#" class="list-group-item list-group-item-action itemclicavel">Gestão de obras</a>
18      <a href="#" class="list-group-item list-group-item-action itemclicavel">Metodologias ativas</a>
19      <a href="#" class="list-group-item list-group-item-action itemclicavel">Projetos mecânicos</a>
20      <a href="#" class="list-group-item list-group-item-action itemclicavel">Processos de fabricação</a>
21    </div>
22  </div>
23 </div>
24 <div class="menubackground"></div>
```

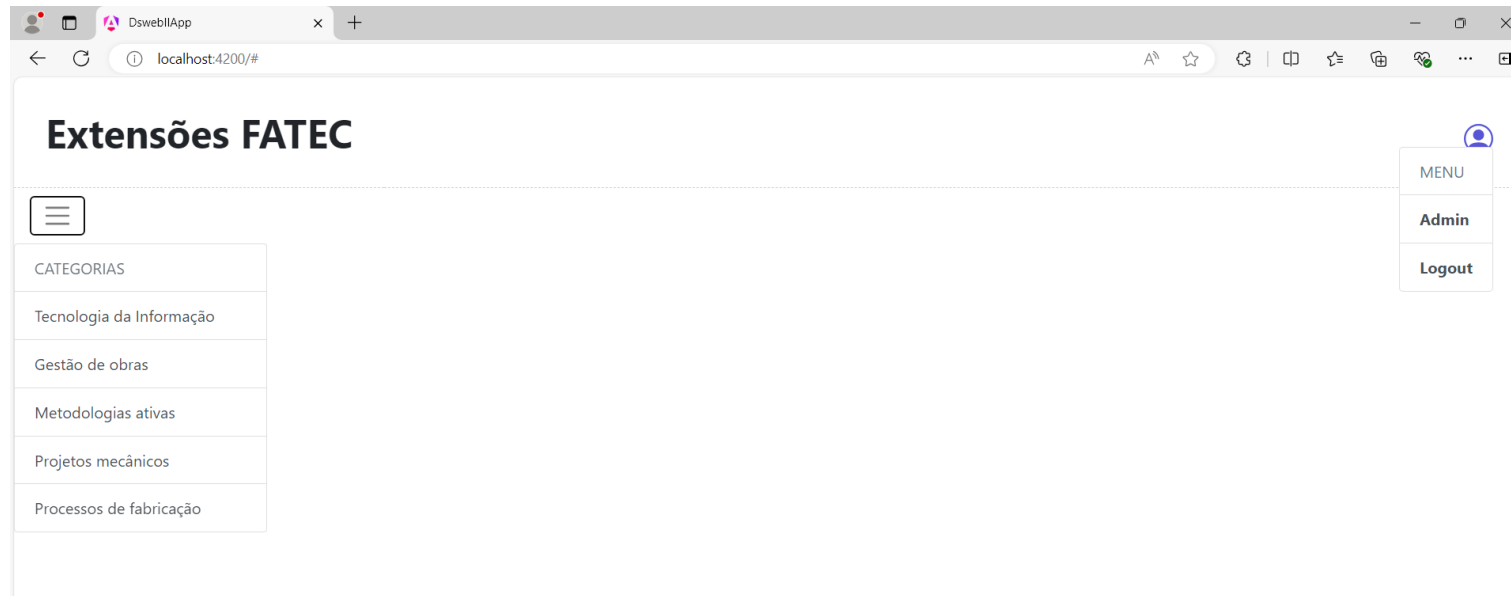
Criando o componente nav-Bar

- Adicione o seguinte css:

```
# nav-bar.component.css U X
1  .menuexpansivel
2  {
3      position: fixed;
4      width: 256px;
5  }
6  .menubackground{
7      width: 100%;
8      height: 55.6px;
9      background-color: white;
10 }
11
```

Criando o componente nav-Bar

- Pronto! Finalizamos o cabeçalho da nossa aplicação.
- Execute novamente a aplicação, verifique o funcionamento dos componentes.



Exercícios

- Reúna-se com seu grupo do Projeto Integrador
- Crie um novo projeto em Angular:
- Adicione os componentes header e a navBar de acordo com o protótipo desenvolvido para seu projeto
- Crie o repositório para a aplicação e publique o código da sua aplicação.
- Anexe o link do repositório, na tarefa disponibilizada no Teams