

1. Components
2. NavbarComponent

- Info
- Source
- Template
- Styles
- DOM Tree

File

src/app/shared/navbar/navbar.component.ts

Description

Component for navbar

Implements

[OnInit](#) [DoCheck](#)

Metadata

selector app-navbar
styleUrls ./navbar.component.scss
templateUrl ./navbar.component.html

Index

Properties

- Public [identity](#)
- Public [photo](#)
- Public [searchForm](#)
- [searchValue](#)
- Public [selectedValue2](#)
- Public [title](#)

Methods

- [logout](#)
- [ngDoCheck](#)
- [ngOnInit](#)
- [searchHP](#)

Constructor

constructor(formBuilder: [FormBuilder](#), _userService: [UserService](#), _router: [Router](#))
Defined in [src/app/shared/navbar/navbar.component.ts:36](#)

Constructor in which we inject our services and different elements

Parameters :

Name	Type	Optional
formBuilder	FormBuilder	No
_userService	UserService	No
_router	Router	No

Methods

logout

logout()
Defined in [src/app/shared/navbar/navbar.component.ts:82](#)

Logout from the app

Returns : [void](#)

ngDoCheck

ngDoCheck()
Defined in [src/app/shared/navbar/navbar.component.ts:74](#)

Called every time that the input properties of a component or a directive are checked. Use it to extend change detection by performing a custom check.

Returns : [void](#)

ngOnInit

ngOnInit()
Defined in [src/app/shared/navbar/navbar.component.ts:55](#)

Start when the component inits

Returns : [void](#)

searchHP

searchHP()
Defined in [src/app/shared/navbar/navbar.component.ts:91](#)

Search hero or user

Returns : [void](#)

Properties

Public [identity](#)

Defined in [src/app/shared/navbar/navbar.component.ts:23](#)

to save the users identity

Public photo

Type : [string](#)

Default value : ''

Defined in [src/app/shared/navbar/navbar.component.ts:36](#)

to save the photo from storage

Public searchForm

Type : [FormGroup](#)

Defined in [src/app/shared/navbar/navbar.component.ts:27](#)

to add FormGroup

searchValue

Type : []

Default value : [{ value: 'HeroName', viewValue: 'Hero' }, { value: 'Marvel', viewValue: 'All Marvel Heroes' }, { value: 'DC', viewValue: 'All DC Heroes' }, { value: 'User', viewValue: 'User' }]

Defined in [src/app/shared/navbar/navbar.component.ts:64](#)

search value options

Public selectedValue2

Type : [string](#)

Default value : 'HeroName'

Defined in [src/app/shared/navbar/navbar.component.ts:31](#)

to save the selected value in the form

Public title

Type : [string](#)

Defined in [src/app/shared/navbar/navbar.component.ts:19](#)

to save the title

```
import { Component, OnInit, DoCheck } from '@angular/core';
import { UserService } from 'src/app/services/user.service';
import { Router } from '@angular/router';
import { FormGroup, FormBuilder, Validators } from '@angular/forms';

/**
 * Component for navbar
 */
@Component({
  selector: 'app-navbar',
  templateUrl: './navbar.component.html',
  styleUrls: ['./navbar.component.scss']
})

export class NavbarComponent implements OnInit, DoCheck {
  /**
   * to save the title
   */
  public title: string;
  /**
   * to save the users identity
   */
  public identity;
  /**
   * to add FormGroup
   */
  public searchForm: FormGroup;
  /**
   * to save the selected value in the form
   */
  public selectedValue2: string = 'HeroName'

  /**
   * to save the photo from storage
   */
  public photo: string = ''

  /**
   * Constructor in which we inject our services and different elements
   */
  constructor(private formBuilder: FormBuilder, private _userService: UserService, private _router: Router) {
    this.searchForm = this.formBuilder.group({
      search: [
        '',
      ],
      selectedValue: [
        ''
      ]
    });
  }

  /**
   * Start when the component inits
   */
  ngOnInit(): void {
    this.identity = this._userService.getIdentity();
    this.photo = localStorage.getItem('UserPhoto')
  }

  /**
   * search value options
   */
  searchValue = [
    { value: 'HeroName', viewValue: 'Hero' },
    { value: 'Marvel', viewValue: 'All Marvel Heroes' },
    { value: 'DC', viewValue: 'All DC Heroes' },
    { value: 'User', viewValue: 'User' }
  ];

  /**
   * Called every time that the input properties of a component or a directive are checked. Use it to extend change detection by performing a custom check.
   */
  ngDoCheck(): void {
    this.identity = this._userService.getIdentity();
    this.photo = localStorage.getItem('UserPhoto')
  }

  /**
   * Logout from the app
   */
  logout() {
    localStorage.clear();
    this.identity = null;
    this._router.navigate(['/']);
  }
}
```

```
}

/**
 * Search hero or user
 */
searchHP() {
  if (this.searchForm.value.search) {
    this._router.navigate(['/heroesSearch/${this.searchForm.value.selectedValue}/${this.searchForm.value.search}']);

  } else {
    this._router.navigate(['/heroesSearch/${this.searchForm.value.selectedValue}/publisher']);
  }
  this.searchForm.reset()
}

}

<nav class="navbar fixed-top navbar-expand-lg navbar-light bg-light">
<svg [routerLink]='["home"]' id="svg8" version="1.1" viewBox="0 0 133.85793 118.58113" height="70">
<defs id="defs2" />
<g transform="translate(-43.277839,-75.016237)" id="layer1" inkscape:groupmode="layer" inkscape:label="Capa 1">
<path id="path117"
d="m 52.719676,184.57778 c 0,-0.1437 0.102394,-0.31383 0.305906,-0.50828 0.475224,-0.45406 1.168202,-1.07327 1.601241,-1.4308 0.216694,-0.17891 0.434317,-0.36264 0.483605,-0.4
style='fill:#7e6e5;fill-opacity:1;stroke-width:0.0813237' />
<path id="path24"
d="m 124.93229,174.54437 c -0.21281,-0.0315 -0.47463,-0.0795 -0.58182,-0.10656 -0.10719,-0.0271 -0.3834,-0.0947 -0.6138,-0.15031 -0.2304,-0.0556 -0.47127,-0.13487 -0.53527,-0.
style='fill:#000000;stroke-width:0.0465455' />
<path id="path47"
d="m 79.194734,168.6879 c -1.022233,-0.56949 -2.168104,-1.52519 -2.168104,-1.80828 0,-0.14365 -0.233766,-0.78976 -0.519481,-1.43581 -0.687656,-1.55492 -0.689102,-2.04445 -0.00
style='fill:#00a23d;fill-opacity:1;stroke-width:0.207792' />
<path id="path49"
d="m 132.26768,168.38492 c -0.0174,-0.6 -0.008,-1.88896 0.021,-2.86436 0.029,-0.9754 -0.0534,-2.47397 -0.18325,-3.33016 -0.19896,-1.31258 -0.1847,-1.70201 0.0909,-2.48322 0.30
style='fill:#00a23d;fill-opacity:1;stroke-width:0.207792' />
<path id="path51"
d="m 132.5339,169.18875 c -0.2905,-0.3618 -0.42846,-8.59545 -0.16162,-9.64621 0.2893,-1.13923 0.0934,-1.28005 -1.78716,-1.28432 -1.34024,-0.003 -1.4026,-0.0253 -1.4026,-0.5007
style='fill:#00a23d;fill-opacity:1;stroke-width:0.207792' />
<path id="path53"
d="m 132.53494,169.18875 c -0.27094,-0.33932 -0.42098,-8.80186 -0.17053,-9.61789 0.32159,-1.04783 -0.0354,-1.31582 -1.75278,-1.31582 -1.36909,0 -1.42911,-0.0209 -1.42911,-0.49
style='fill:#00a23d;fill-opacity:1;stroke-width:0.207792' />
<path id="path55"
d="m 132.53494,169.18875 c -0.27094,-0.33932 -0.42098,-8.80186 -0.17053,-9.61789 0.32159,-1.04783 -0.0354,-1.31582 -1.75278,-1.31582 -1.36909,0 -1.42911,-0.0209 -1.42911,-0.49
style='fill:#00a23d;fill-opacity:1;stroke-width:0.207792' />
<path id="path57"
d="m 122.4589,171.91739 c -0.81733,-0.74486 -2.46611,-6.76526 -2.10028,-7.66898 0.14879,-0.36755 5.72573,-3.49407 7.05767,-3.95665 0.57142,-0.19844 1.39885,-0.36386 1.83872,-0
style='fill:#00a23d;fill-opacity:1;stroke-width:0.207792' />
<path id="path59"
d="m 122.6862,171.91739 c -0.47518,-0.36091 -1.12666,-1.9771 -1.72456,-4.27828 -0.77752,-2.99251 -0.74082,-3.21271 0.68841,-4.13055 0.65714,-0.42201 2.3389,-1.35591 3.73725,-2
style='fill:#00a23d;fill-opacity:1;stroke-width:0.207792' />
<path id="path61"
d="m 122.69843,171.92953 c -0.48745,-0.37415 -1.13466,-1.97296 -1.73679,-4.29042 -0.77752,-2.99251 -0.74082,-3.21271 0.68841,-4.13055 0.65714,-0.42201 2.3389,-1.35591 3.73725,
style='fill:#00a23d;fill-opacity:1;stroke-width:0.207792' />
<path id="path63"
d="m 122.5852,171.92814 c -0.69979,-0.8432 -2.12996,-5.50655 -2.12996,-6.94518 0,-0.63333 0.12465,-0.78716 1.19481,-1.4744 0.65714,-0.42201 2.3389,-1.35591 3.73725,-2.07534 2.
style='fill:#00a23d;fill-opacity:1;stroke-width:0.207792' />
<path id="path65"
d="m 122.4584,171.73047 c -0.75893,-1.1602 -2.00316,-5.35296 -2.00316,-6.75021 0,-0.64336 0.12157,-0.78774 1.2987,-1.54236 0.71429,-0.45791 2.40435,-1.3906 3.75568,-2.07264 2.
style='fill:#00a23d;fill-opacity:1;stroke-width:0.207792' />
<path transform="scale(0.26458333)" id="path69" d="" style="fill:#00a23d;fill-opacity:1;stroke-width:0.0892857" />
<path id="path85"
d="m 48.711766,161.12795 c -1.518955,-0.63841 -1.497648,-0.5819 -2.776668,-7.36715 -0.598077,-3.17283 -0.657744,-5.46873 -0.180314,-6.93839 1.272474,-3.91699 6.140039,-8.53045
style='fill:#f4d403;fill-opacity:1;stroke-width:0.171915' />
<text id="text24" y="192.97302" x="100.05083"
style="font-style:normal;font-weight:normal;font-size:10.5833px;line-height:1.25;font-family:sans-serif;fill:#cc4224;fill-opacity:1;stroke:none;stroke-width:0.264583;"
xml:space="preserve">
<tspace
style="font-style:normal;font-variant:normal;font-weight:normal;font-stretch:normal;font-size:25.4px;font-family:Bangers;-inkscape-font-specification:Bangers;stroke-width:0.
y="192.97302" x="100.05083" id="tspan22" sodipodi:role="line">My Hero</tspace>
</text>
</g>
</svg>

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarSupportedContent"
aria-controls="navbarSupportedContent" aria-expanded="false" aria-label="Toggle navigation">
<span class="navbar-toggler-icon"></span>
</button>

<div class="collapse navbar-collapse" id="navbarSupportedContent">
<ul class="navbar-nav mr-auto">
<li class="nav-item">
<a class="nav-link" [routerLink]='["home"]' routerLinkActive="active">HOME</a>
</li>
<li class="nav-item">
<a class="nav-link" [routerLink]='["fight/fightHero"]' routerLinkActive="active">FIGHT</a>
</li>
<li class="nav-item">
<a class="nav-link" [routerLink]='["myHeroes"]' routerLinkActive="active">MY HEROES</a>
</li>
<li class="nav-item" *ngIf="!identity">
<a class="nav-link" [routerLink]='["login"]' routerLinkActive="active">LOGIN</a>
</li>
<li class="nav-item" *ngIf="!identity">
<a class="nav-link" [routerLink]='["register"]' routerLinkActive="active">REGISTER</a>
</li>
</ul>

<div id="form" class="mr-auto fromG">
<form [formGroup]="searchForm" (ngSubmit)="searchHP()">
<mat-form-field appearance="fill" class="selectO">
<mat-label>Select</mat-label>
<mat-select [(ngModel)]="selectedValue2" formControlName="selectedValue" name="searchValue">
<mat-option *ngFor="let sel of searchValue2" [value]="sel.value">
{{ sel.viewValue }}
</mat-option>
</mat-select>
</mat-form-field>

<mat-form-field appearance="fill" *ngIf="selectedValue2 === 'HeroName'">
<!-- <mat-label>Search</mat-label> -->
<!-- <fieldset [disabled]="selectedValue2 != 'HeroName'" -->
<input matInput #input maxlength="20" placeholder="{{ selectedValue2 }}" formControlName="search" />
<!-- </fieldset -->
</mat-form-field>

<mat-form-field appearance="fill" *ngIf="selectedValue2 === 'User'">
<!-- <mat-label>Search</mat-label> -->
<!-- <fieldset [disabled]="selectedValue2 != 'HeroName'" -->
<input matInput #input maxlength="20" placeholder="{{ selectedValue2 }}" formControlName="search" />
<!-- </fieldset -->
</mat-form-field>

<button type="submit" id="search-button">

<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 512 512">
<path fill="#7d7ebc" stroke="#000" stroke-miterlimit="10" stroke-width="15"
d="M63,300.3A173.58,173.58,0,1,0,71.55,55,173.59,173.59,0,0,0,63,300.32" />
<circle cx="189.96" cy="181.98" r="139.72" fill="#78d1f1" stroke="#000" stroke-miterlimit="10"
stroke-width="15" transform="rotate(-43.01 189.951 181.978)" />
<path fill="#7d7ebc" stroke="#000" stroke-miterlimit="10" stroke-width="15"
d="M333.41,373.93138.06-35.5L489.38,460.31a26,26,0,0,1-1.28,36.78h0a26,26,0,0,1-36.78-1.282" />
<line x1="352.47" x2="308.34" y1="356.15" y2="308.84" fill="none" stroke="#000" stroke-miterlimit="10"
stroke-width="15" />
<path fill="none" stroke="#fff" stroke-linecap="round" stroke-miterlimit="10" stroke-width="15"
d="M238.65,103.16A103.19,103.19,0,0,0,98,141.92" /></svg>
</button>
</form>
</div>
```

```
<ul class="nav navbar-nav navbar-right mr-4" *ngIf="identity">
  <!-- <li>
    <img class="imgRedonda" [src]="identity.photo" [alt]="identity.alias" />
  </li> -->
  <li class="dropdown">
    <img class="userAvatar" [src]="this.photo" [alt]="identity.alias" />

    <a class="dropdown-toggle name" data-toggle="dropdown" href="#">
      {{ identity.alias }} <span class="caret"></span>
    </a>
    <div class="dropdown-menu" aria-labelledby="navbarDropdown">
      <a class="dropdown-item" [routerLink]="['profile']">Profile</a>
      <a *ngIf="identity.admin == 1" class="dropdown-item" [routerLink]="['admin-panel']">Admin</a>
      <a (click)="logout()" class="dropdown-item" href="#">Exit</a>
    </div>
  </li>
</ul>
</div>
</nav>

./navbar.component.scss

.mr-4,
.mx-4 {
  margin-right: 5.5rem;
}

.bg-light {
  background-color: #272727 !important;
}

.nav-link {
  font-family: "B612";
  color: rgb(223, 221, 221) !important;
  font-weight: 900;
  font-size: 15px;
}
.nav-link:hover {
  color: #cc4224 !important;
}

.active {
  color: #00a23d !important;

  font-weight: 500;
}
mat-form-field {
  margin-right: 0.2em;
}

::ng-deep.mat-form-field-appearance-fill .mat-form-field-infix {
  padding: 0.15em 0 0.15em 0 !important;
}

::ng-deep.mat-form-field-appearance-fill .mat-form-field-flex {
  // background: rgba(0, 255, 255, .5);
  background: rgb(223, 221, 221);
  width: 8em !important;
  border-radius: 10px 10px 10px 10px !important;
  padding: 0.35em 0.35em 0 0.35em !important;
}
.fromG {
  margin-top: 1em;
}

.fixed-top {
  z-index: 20 !important;
}

#search-button {
  position: absolute;
  background: transparent;
  border: none !important;
}

#search-button svg {
  height: 34px;
}

.name {
  color: rgb(223, 221, 221);
  font-family: "B612";
}
a {
  font-family: "B612";
}

.navbar-toggler {
  background-color: #8d2529;
  border: 2px solid;
}

.navbar-light .navbar-toggler {
  border-color: #00a23d;
}

#svg8 {
  margin-right: 2em;
  cursor: pointer;
}

@media (min-width: 992px) {
  .navbar-expand-lg .navbar-collapse {
    height: 2em;
  }

  mat-form-field {
    margin-right: 1em;
  }

  ::ng-deep.mat-form-field-appearance-fill .mat-form-field-flex {
    width: 10em !important;
  }
}
@media (min-width: 768px) and (max-width: 991px) {
  .mr-4,
  .mx-4 {
    margin-right: 1.5rem !important;
  }
}
```

Legend
 Html element
 Component
 Html element with directive

result-matching ""

No results matching ""