

- 1. Components
- 2. TeamComponent

- [Info](#)
- [Source](#)
- [Template](#)
- [Styles](#)
- [DOM Tree](#)

File

src/app/components/team/team.component.ts

Description

Component that brings team info and details to export

Implements

[OnInit](#)

Metadata

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

Index

Properties

- Public [countM](#)
- Public [i](#)
- Public [myTeamInfo](#)
- Public [TeamHeroes](#)

Methods

- [countMembers](#)
- [deleteMember](#)
- [getMemberNull](#)
- [getTeamUsu](#)
- [ngOnChanges](#)
- [ngOnInit](#)
- [teamUsugetHeroes](#)

Inputs

- [idUsu](#)
- [newM](#)
- [showName](#)
- [type](#)

Constructor

constructor(\_TeamService: [TeamService](#), \_heroService: [HeroService](#), GlobalV: [GlobalVariableService](#))  
Defined in [src/app/components/team/team.component.ts:43](#)

Constructor in which we inject team service, hero service and our Global variables

**Parameters :**

Name	Type	Optional
_TeamService	<a href="#">TeamService</a>	No
_heroService	<a href="#">HeroService</a>	No
GlobalV	<a href="#">GlobalVariableService</a>	No

## Inputs

**idUsu**

Type : [number](#)

Default value : 0

Defined in [src/app/components/team/team.component.ts:40](#)

**newM**

Defined in [src/app/components/team/team.component.ts:41](#)

**showName**

Defined in [src/app/components/team/team.component.ts:43](#)

**type**

Defined in [src/app/components/team/team.component.ts:42](#)

## Methods

**countMembers**

countMembers()

Defined in [src/app/components/team/team.component.ts:126](#)

Count the members in the club

Returns : [void](#)

**deleteMember**

deleteMember(number)

Defined in [src/app/components/team/team.component.ts:109](#)

Delete a team member

**Parameters :**

Name Optional

number No

Returns : [void](#)

**getMemberNull**

getMemberNull()

Defined in [src/app/components/team/team.component.ts:138](#)

Get the member that is null

Returns : [any](#)

member

**getTeamUsu**

getTeamUsu()

Defined in [src/app/components/team/team.component.ts:70](#)

Get the team information

Returns : [void](#)

**ngOnChanges**

ngOnChanges()

Defined in [src/app/components/team/team.component.ts:63](#)

Activates when it detects any change

Returns : [void](#)

**ngOnInit**

ngOnInit()

Defined in [src/app/components/team/team.component.ts:57](#)

Start when the component inits

**Returns :** [void](#)

#### **teamUsugetHeroes**

teamUsugetHeroes(member, position)

Defined in [src/app/components/team/team.component.ts:99](#)

Get the heroes and put them in the team array

#### **Parameters :**

Name    Optional

member No

position No

**Returns :** [void](#)

### **Properties**

#### **Public countM**

Type : [boolean](#)

Default value : false

Defined in [src/app/components/team/team.component.ts:36](#)

Variable to count team members

#### **Public i**

Type : [number](#)

Default value : 0

Defined in [src/app/components/team/team.component.ts:38](#)

#### **Public myTeamInfo**

Type : [Team](#)

Defined in [src/app/components/team/team.component.ts:28](#)

variable to save the team info

#### **Public TeamHeroes**

Type : [Hero\[\]](#)

Default value : []

Defined in [src/app/components/team/team.component.ts:32](#)

variable to save the team members

```
import {
  Component,
  OnInit,
  Input,
  SimpleChanges,
  OnChanges,
  DoCheck
} from '@angular/core'
import { TeamService } from 'src/app/services/team.service'
import { HeroService } from 'src/app/services/hero.service'
import { Team } from 'src/app/models/team'
import { Hero } from 'src/app/models/hero'
import { GlobalVariableService } from 'src/app/services/global-variable.service'

/**
 * Component that brings team info and details to export
 */
@Component({
  selector: 'app-team',
  templateUrl: './team.component.html',
  styleUrls: ['./team.component.scss']
})

export class TeamComponent implements OnInit {
  /**
   * variable to save the team info
   */
  public myTeamInfo: Team
  /**
   * variable to save the team members
   */
  public TeamHeroes: Hero[] = []
  /**
   * Variable to count team members
```

```

    */
    public countM: boolean = false

    public i: number = 0;
    // Decorator that will receive the id, with the input we define when this component is called, and an id must be sent to it
    @Input() idUsu: number = 0
    @Input() newM
    @Input() type
    @Input() showName

    /**
     * Constructor in which we inject team service, hero service and our Global variables
     */
    constructor(
        private _TeamService: TeamService,
        private _heroService: HeroService,
        private GlobalV: GlobalVariableService
    ) { }

    /**
     * Start when the component inits
     */
    ngOnInit() {
        this.getTeamUsu()
    }
    /**
     * Activates when it detects any change
     */
    ngOnChanges() {
        this.getTeamUsu()
    }

    /**
     * Get the team information
     */
    getTeamUsu(): void {
        this._TeamService.getTeamInfo(this.idUsu).subscribe(
            res => {
                this.myTeamInfo = res
                if (this.myTeamInfo) {
                    // para saber si ya se han countM los miembros del equipo
                    if (this.countM === false) {
                        this.countMembers()
                    }
                    this.countM = true
                    this.GlobalV.memberTeamNull = this.getMemberNull()
                    this.teamUsugetHeroes(this.myTeamInfo.member_1, 0)
                    this.teamUsugetHeroes(this.myTeamInfo.member_2, 1)
                    this.teamUsugetHeroes(this.myTeamInfo.member_3, 2)
                    this.teamUsugetHeroes(this.myTeamInfo.member_4, 3)
                    this.teamUsugetHeroes(this.myTeamInfo.member_5, 4)
                }
            },
            error => {
                console.log(error)
            }
        )
    }

    /**
     * Get the heroes and put them in the team array
     * @param {string} member
     * @param {number} position
     */
    teamUsugetHeroes(member, position): void {
        this._heroService.getHeroById(member).subscribe(res => {
            this.TeamHeroes[position] = res
        })
    }

    /**
     * Delete a team member
     * @param {number} number
     */
    deleteMember(number): void {
        var memberSelect = `member_${number}`
        var data = { member: memberSelect }
        this._TeamService.deleteMember(this.myTeamInfo.idTeam, data).subscribe(
            res => {
                this.getTeamUsu()
                this.GlobalV.countTeamMembers--
            },
            error => {
                console.log(error)
            }
        )
    }

    /**
     * Count the members in the club
     */
    countMembers(): void {
        for (var i = 1; i < 6; i++) {
            if (this.myTeamInfo[`member_${i}`] != null) {
                this.GlobalV.countTeamMembers++
            }
        }
    }

```

```

    }
  }
}

/**
 * Get the member that is null
 * @returns member
 */
getMemberNull(): any {
  // we check the member slot that is empty and store it
  for (var i = 1; i < 6; i++) {
    if (this.myTeamInfo[member_${i}] == null) {
      return member_${i}
    }
  }
}
}
}

<h3 *ngIf="myTeamInfo && showName" class="teamName">
  TEAM NAME: <span>{{ myTeamInfo.teamName }}</span>
</h3>
<div id="TeamContainer">
  <div class="card3 tWiner" *ngFor="let hero of TeamHeroes; index as i">
    

    <img
      *ngIf="hero"
      class="img"
      [src]="hero.image"
      [routerLink]="['/heroesDetail', hero.idHero]"
      [alt]="hero.heroName"
    />

    
    <div class="infoHero">
      <h4 *ngIf="hero">{{ hero.heroName }}</h4>
    </div>
    <div class="semiCircle"></div>
  </div>
</div>

./team.component.scss

#TeamContainer {
  display: flex;
  flex-wrap: wrap;
  justify-content: space-around;
}

.card3 {
  font-family: "B612";
  margin: 1em;
  display: flex;
  height: 55vw;
  flex-direction: column;
  align-items: center;
  justify-content: center;
  text-align: center;
  box-shadow: 0 0 21px 9px rgba(8, 17, 10, 0.63);
  width: 40%;
  background-color: #272727;
  overflow: hidden;
  position: relative;
  border-radius: 2%;
}

.card3 .imgCircle {
  cursor: pointer;
  margin-bottom: 5em;
  border-radius: 50%;
  overflow: hidden;
}

.semiCircle {
  width: 138%;
  height: 73%;
  bottom: -48%;
  z-index: 0;
  position: absolute;
  background: #00a23d;
  border-radius: 201px 201px 0 0;
  display: flex;
  flex-direction: row;
  justify-content: space-around;
}

.infoHero {
  font-family: "B612";

```

```

    height: 40%;
    width: 81%;
    z-index: 2;
    bottom: -25%;
    color: rgb(223, 221, 221);
    position: absolute;
}

.card3:hover img {
    cursor: pointer;
    transform: scale(1.1);
    -ms-transform: scale(1.1);
    -moz-transform: scale(1.1);
    -webkit-transform: scale(1.1);
    -o-transform: scale(1.1);
}

.card3:hover .semiCircle {
    background: #cc4224;
}

.card3 .img {
    height: 100%;
    width: 100%;
}

.picsum-img-wrapper img {
    width: 100%;
}

.delteMember {
    z-index: 2;
    bottom: 0em;
    position: absolute;
}

.card3 h4 {
    font-size: 4vw;
}

.iconBuuton {
    cursor: pointer;
    height: 6vw;
    z-index: 4;
    right: 2vw;
    top: 47.5vw;
    position: absolute;
}

.teamName {
    color: #cc4224;
    font-family: "B612";
}

.teamName span {
    color: #00a23d;
}

@media (min-width: 992px) {
    #TeamContainer {
        flex-direction: row;
    }

    .iconBuuton {
        height: 1.5vw;
    }

    .card3 h4 {
        font-size: 1.2vw;
    }

    .img-fluid {
        width: 100%;
    }

    .card3 {
        width: 16%;
        height: 18vw;
    }

    .iconBuuton {
        cursor: pointer;
        height: 1.5vw;
        z-index: 4;
        right: 1vw;
        top: 16vw;
        position: absolute;
    }
}

```

## Legend

Html element

Component

Html element with directive

**result-matching ""**

**No results matching ""**