

- 1. Components
- 2. UserListComponent

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

File

src/app/admin/components/user-list/user-list.component.ts

Description

Component that brings the List of users

Implements

[OnInit](#)

Metadata

selector app-user-list
styleUrls ./user-list.component.scss
templateUrl ./user-list.component.html

Index

Properties

- Public [dialog](#)
- [page](#)
- [pageSize](#)
- Public [users](#)

Methods

- [deleteUserDialog](#)
- [deteteUser](#)
- [getAllUsers](#)
- [ngOnInit](#)

Constructor

constructor(dialog: MatDialog, _userService: [UserService](#))
Defined in [src/app/admin/components/user-list/user-list.component.ts:22](#)

Constructor in which we inject our services and different elements

Parameters :

Name	Type	Optional
dialog	MatDialog	No
_userService	UserService	No

Methods

deleteUserDialog

deleteUserDialog(idUsu, name)

Defined in [src/app/admin/components/user-list/user-list.component.ts:64](#)

Open modal to delete a user

Parameters :

Name Optional

idUsu No

name No

Returns : [void](#)

deteteUser

deteteUser(idUsu)

Defined in [src/app/admin/components/user-list/user-list.component.ts:51](#)

To delete a user from app

Parameters :

Name Optional

idUsu No

Returns : [void](#)

getAllUsers

getAllUsers()

Defined in [src/app/admin/components/user-list/user-list.component.ts:39](#)

To get all users in the app

Returns : [void](#)

ngOnInit

ngOnInit()

Defined in [src/app/admin/components/user-list/user-list.component.ts:32](#)

Start when the component inits

Returns : [void](#)

Properties

Public dialog

Type : MatDialog

Defined in [src/app/admin/components/user-list/user-list.component.ts:27](#)

page

Type : [number](#)

Default value : 1

Defined in [src/app/admin/components/user-list/user-list.component.ts:17](#)

pageSize

Type : [number](#)

Default value : 10

Defined in [src/app/admin/components/user-list/user-list.component.ts:18](#)

Public users

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

Default value : []

Defined in [src/app/admin/components/user-list/user-list.component.ts:22](#)

variable to store the list of users from app

```
import { Component, OnInit } from '@angular/core';
import { User } from 'src/app/models/user';
import { UserService } from 'src/app/services/user.service';
import { MatDialog } from "@angular/material/dialog";
import { AdminDeleteUserDialogComponent } from 'src/app/components/modals/admin-delete-user-dialog/admin-delete-user-dialog.component';

/**
 * Component that brings the List of users
 */
@Component({
  selector: 'app-user-list',
  templateUrl: './user-list.component.html',
  styleUrls: ['./user-list.component.scss']
})

export class UserListComponent implements OnInit {
  page = 1;
  pageSize = 10;
  /**
   * variable to store the list of users from app
   */
  public users: User[] = [];

  /**
   * Constructor in which we inject our services and different elements
   */
}
```

```

constructor(public dialog: MatDialog, private _userService: UserService) { }

/**
 * Start when the component inits
 */
ngOnInit() {
  this.getAllUsers()
}

/**
 * To get all users in the app
 */
getAllUsers() {
  this._userService.allusers().subscribe(res => {
    this.users = res;
  }, err => {
    console.log(err)
  })
}

/**
 * To delete a user from app
 * @param {number} idUsu
 */
deteteUser(idUsu) {
  this._userService.adminDeleteUser(idUsu).subscribe(res => {
    this.getAllUsers()
  }, err => {
    console.log(err)
  })
}

/**
 * Open modal to delete a user
 * @param {number} idUsu
 * @param {string} name
 */
deleteUserDialog(idUsu, name): void {
  const dialogRef = this.dialog.open(AdminDeleteUserDialogComponent, {
    data: {
      userId: idUsu,
      UserName: name
    }
  });
  dialogRef.afterClosed().subscribe(result => {
    console.log("The dialog was closed");
    if (result) {
      this.deteteUser(result)
    }
  });
}
}

<div class="adminDetail">
  <div class="title">
    <h1>USERS LIST</h1>
  </div>

  <div class="tablaContainer">
    <table class="table">
      <thead>
        <tr>
          <th class="imgUser" scope="col"></th>
          <th scope="col">ALIAS</th>
          <th scope="col" class="nodisplay">CREATE</th>
          <th scope="col" class="nodisplay">EMAIL</th>
          <th scope="col">DELETE</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <td>
            *ngFor="
              let user of users
              | slice: (page - 1) * pageSize:(page - 1) * pageSize + pageSize;
              index as i
            "
          >
            <td class="usu imgUser">
              <div class="userAvatar2 img">
                
              </div>
            </td>
            <td class="textName" [routerLink]="['/publicProfile', user.idUsu]">
              <p>
                {{
                  user.alias.length > 20
                  ? (user.alias | slice: 0:20) + "..."
                  : user.alias
                }}
              </p>
            </td>
            <td class="nodisplay">
              <p>
                {{ user.createDate | date }}
              </p>
            </td>
            <td class="nodisplay">
              <p>

```

```

        {{ user.email }}
      </p>
    </td>
  </tr>
</tbody>
</table>
</div>

<!-- <button type="button" mat-raised-button (click)="deleteUserDialog(user.idUsu,user.alias)">Delete
      user</button> -->
    </td>
  </tr>
</tbody>
</table>
</div>

<!-- </div> -->
<div class="pag">
  <ngb-pagination
    [(page)]="page"
    [pageSize]="pageSize"
    [collectionSize]="users.length"
  >
</ngb-pagination>
</div>
</div>
<!-- <div *ngFor=" let user of users">
  <h3>{{ user.alias }}</h3>
  <br />
  <br />
  <img [src]="user.photo" [alt]="user.alias" class="img-fluid" />
  <button type="button" mat-raised-button (click)="deleteUserDialog(user.idUsu,user.alias)">Delete user</button>
</div> -->
<!-- </div> -->

./user-list.component.scss

.adminDetail {
  border: 2.5px solid #00a23d;
  box-shadow: 0 0 21px 9px rgba(8, 17, 10, 0.63);
  padding: 0.5em;
  margin: 0.5em;
  background-color: #272727;
  color: white;
  margin-bottom: 3em;
}

.modifyHero {
  outline: none;
  background-color: #f3d403;
  border: none;
  border-radius: 10px;
  border: 2.5px solid #00a23d;
  padding: 9px;
  font-size: 0.8rem;
  color: black;
}

.modifyHero:hover {
  background-color: #cc4224;
  color: rgb(223, 221, 221);
}

.title {
  display: flex;
  flex-direction: column;
}

///table
.tablaContainer {
  // width: ;
  color: #00a23d;
}

.userAvatar2 {
  height: 4em;
  width: 4em;
  background-repeat: no-repeat;
  background-position: 50%;
  border-radius: 50%;
  background-size: 100% auto;
  overflow: hidden;
  border: #00a23d 2px solid;
  height: auto;
  width: 4.1em;
  height: 4.1em;
}

.imgUser {
  height: auto;
  width: 4.1em;
  height: 4.1em;
}

.table tr {
  color: #00a23d;
}

.table td {
  color: rgb(223, 221, 221);
}

```

```

.star2 {
  // margin-top: -1.2em;
  font-size: 15px;
  color: #cc4224;
  outline: none;
}

.pag {
  display: flex;
  align-items: center;
  justify-content: center;
  margin: 1em 0 1em 0;
}

.textName {
  padding-top: 30px !important;
  cursor: pointer;
}

.textName:hover p {
  color: #cc4224;
}

//pagination
ngb-pagination ::ng-deep a {
  color: black !important;
}

ngb-pagination ::ng-deep .page-item.active .page-link {
  color: white !important;
  border-color: black !important;
  background-color: #00a23d;
}

ngb-pagination ::ng-deep .page-item.active .page-link:focus,
.page-link:not(:disabled):not(.disabled):active:focus {
  box-shadow: 0 0 0 0.1rem black;
}

ngb-pagination ::ng-deep .page-link:not(:disabled):not(.disabled):active:focus {
  box-shadow: 0 0 0 0.1rem black;
}

.nodisplay {
  display: none;
}

.iconBuuton {
  cursor: pointer;
  height: 3em;
}

@media (min-width: 992px) {
  .title {
    display: flex;
    flex-direction: row;
  }
  .nodisplay {
    display: table-cell;
  }
  .adminDetail {
    margin: 3em;
    padding: 2em;
  }
}

```

- Legend**
- Html element
 - Component
 - Html element with directive

result-matching ""

No results matching ""