

# Projeto de Interfaces WEB

## **Projeto-Crud** **Aula 10**

# Introdução

- O objetivo desse mini-projeto é:
  - Criar uma aplicação em Angular, que faça as operações CRUD em um servidor fake JSON

# Criando o projeto

```
└─ app
  └─ core
    ├── edit-user
    ├── list-user
    ├── login-user
    └── register-user
  TS core.module.ts
  └─ models
    TS User.ts
  └─ services
    TS user.service.ts
  # app.component.css
  <> app.component.html
  TS app.component.spec.ts
  TS app.component.ts
  TS app.module.ts
```

- 1 – Crie o módulo “core”
- 2 – Crie os componentes “edit-user”, “list-user”, “register-user” e “login-user” dentro do módulo “core”.
- 3 – Crie a pasta “models” e o arquivo “User.ts” dentro dela.
- 4 – Crie a pasta “service” e o serviço “user” dentro dela.
- 5 – Crie na raiz do projeto uma pasta “data” e um arquivo “db.json” dentro da pasta “data”.
- 6 – O arquivo “db.json” terá o seguinte conteúdo:

```
{
  "users": [
    {
      "login": "ful",
      "firstName": "Fulano da Silva",
      "email": "fulano@gmail.com",
      "zipcode": "60040-562",
      "lastName": "Sales",
      "password": "456123",
      "id": 2
    }
  ]
}
```

# User.ts

```
export class User{  
  id:number;  
  login:string;  
  firstName:string;  
  lastName:string;  
  email:string;  
  zipcode:string;  
  password:string;  
}
```

# user.service.ts

```
import { User } from '../models/User';
import { Injectable } from '@angular/core';
import { HttpClient } from '@angular/common/http';

@Injectable({
  providedIn: 'root'
})
export class UserService {

  url:string = "http://localhost:3000/users";

  constructor(private httpClient:HttpClient) { }

  register(user:User){
    return this.httpClient.post(this.url,user); //Observable
  }

  list(){
    return this.httpClient.get(this.url); //Observable
  }
}
```

# user.service.ts (cont)

```
delete(id:number){  
  return this.httpClient.delete(`${this.url}/${id}`);  
}  
  
retrieveById(id:number){  
  return this.httpClient.get(`${this.url}/${id}`);  
}  
  
update(user:User){  
  return this.httpClient.put(`${this.url}/${user.id}`,user);  
}  
  
retrieveByLogin(login:string){  
  return this.httpClient.get(`${this.url}?login=${login}`);  
}  
}
```

# list-user.component.html

```
<h2>List Users</h2>
```

```
<table class="table table-dark">
```

```
  <thead>
```

```
    <tr>
```

```
      <th scope="col">ID</th>
```

```
      <th scope="col">Name</th>
```

```
      <th scope="col"></th>
```

```
    </tr>
```

```
  </thead>
```

```
  <tbody>
```

```
    <tr *ngFor="let u of users">
```

```
      <th scope="row">{{u.id}}</th>
```

```
      <td>{{u.firstName}}</td>
```

```
      <td align="right">
```

```
        <button class="btn btn-primary" (click)="edit(u.id)">Edit</button>
```

```
        <button class="btn btn-danger ml-3" (click)="delete(u.id)">Delete</button>
```

```
      </td>
```

```
    </tr>
```

```
  </tbody>
```

```
</table>
```

# list-user.component.ts

```
import { UserService } from '../services/user.service';
import { User } from '../models/User';
import { Component, OnInit } from '@angular/core';
import { Router } from '@angular/router';

@Component({
  selector: 'app-list-user',
  templateUrl: './list-user.component.html',
  styleUrls: ['./list-user.component.css']
})
export class ListUserComponent implements OnInit {

  users:User[];

  constructor(private userService:UserService,
               private router:Router) {
    this.users = [];
  }

  ngOnInit() {
    this.list();
  }
}
```



# list-user.component.ts (cont)

```
list(){
  this.userService.list().subscribe(
    (res:User[])=>{
      this.users = res;
    }
  );
}

delete(id:number){
  this.userService.delete(id).subscribe(
    (res:any)=>{
      console.log(`User id ${id} deleted!`);
      this.list();
    }
  );
}

edit(id:number){
  //nevegação imperativa
  this.router.navigate(['edit/user',id]);
}
```

# register-user.component.ts

```
import { UserService } from '../services/user.service';
import { User } from '../models/User';
import { Component, OnInit } from '@angular/core';

@Component({
  selector: 'app-register-user',
  templateUrl: './register-user.component.html',
  styleUrls: ['./register-user.component.css']
})
export class RegisterUserComponent implements OnInit {

  user: User;

  constructor(private userService: UserService) {
    this.user = new User();
  }

  ngOnInit() {
  }

  onSubmit(){
    this.userService.register(this.user).subscribe(
      (res: User) => {
        console.log(`User id ${res.id} added!`)
      }
    );
  }
}
```

# register-user.component.html

```
<h2>Register User</h2>
<hr>
<form (ngSubmit)="onSubmit()">

  <div class="form-group">
    <label for="login">Login</label>
    <input type="text" class="form-control" name="login" id="login"
      [(ngModel)]="user.login">
    </div>

    <div class="form-group">
      <label for="firstName">First Name</label>
      <input type="text" class="form-control" name="firstName" id="firstName"
        [(ngModel)]="user.firstName">
    </div>

    <div class="form-group">
      <label for="lastName">Last Name</label>
      <input type="text" class="form-control" name="lastName" id="lastName"
        [(ngModel)]="user.lastName">
    </div>
  </form>
```

# register-user.component.html (cont)

```
<div class="form-group">
  <label for="email">E-mail</label>
  <input type="text" class="form-control" name="email" id="email"
    [(ngModel)]="user.email">
</div>

<div class="form-group">
  <label for="zipcode">Zipcode</label>
  <input type="text" class="form-control" name="zipcode" id="zipcode"
    [(ngModel)]="user.zipcode">
</div>

<div class="form-group">
  <label for="password">Password</label>
  <input type="password" class="form-control" name="password" id="password"
    [(ngModel)]="user.password">
</div>

<button type="submit" class="btn btn-primary">Register</button>

</form>
```

# edit-user.component.ts

```
import { UserService } from '../services/user.service';
import { User } from '../models/User';
import { Component, OnInit } from '@angular/core';
import { ActivatedRoute, Router } from '@angular/router';
```

```
@Component({
  selector: 'app-edit-user',
  templateUrl: './edit-user.component.html',
  styleUrls: ['./edit-user.component.css']
})
export class EditUserComponent implements OnInit {
```

```
  user: User = new User();
```

```
  constructor(private activatedRoute: ActivatedRoute,
               private userService: UserService,
               private router: Router) { }
```

```
  ngOnInit() {
    let id = this.activatedRoute.snapshot.params["id"];
    //console.log("ID "+id);
```

# edit-user.component.ts (cont)

```
this.userService.retrieveById(id).subscribe(
  (res:User)=>{
    //console.log(res);
    this.user = res;
  }
);
}

onSubmit(){
  this.userService.update(this.user).subscribe(
    (res:User)=>{
      console.log(`User id ${res.id} updated!`);
      this.router.navigate(["list/user"]);
    },
  );
}
}
```

# edit-user.component.html

```
<h2>Edit User</h2>
<hr>
<form (ngSubmit)="onSubmit()">

  <div class="form-group">
    <label for="login">Login</label>
    <input type="text" class="form-control" name="login" id="login"
      [(ngModel)]="user.login">
    </div>

    <div class="form-group">
      <label for="firstName">First Name</label>
      <input type="text" class="form-control" name="firstName" id="firstName"
        [(ngModel)]="user.firstName">
    </div>

    <div class="form-group">
      <label for="lastName">Last Name</label>
      <input type="text" class="form-control" name="lastName" id="lastName"
        [(ngModel)]="user.lastName">
    </div>
```

# edit-user.component.html (cont)

```
<div class="form-group">  
  <label for="email">E-mail</label>  
  <input type="text" class="form-control" name="email" id="email"  
    [(ngModel)]="user.email">  
</div>
```

```
<div class="form-group">  
  <label for="zipcode">Zipcode</label>  
  <input type="text" class="form-control" name="zipcode" id="zipcode"  
    [(ngModel)]="user.zipcode">  
</div>
```

```
<div class="form-group">  
  <label for="password">Password</label>  
  <input type="password" class="form-control" name="password" id="password"  
    [(ngModel)]="user.password">  
</div>
```

```
<button type="submit" class="btn btn-primary">Update</button>
```

```
</form>
```



# login-user.component.ts

```
import { Router } from '@angular/router';
import { UserService } from '../../../services/user.service';
import { User } from '../../../models/User';
import { Component, OnInit } from '@angular/core';
import { ToastrService } from 'ngx-toastr';
```

```
@Component({
  selector: 'app-login-user',
  templateUrl: './login-user.component.html',
  styleUrls: ['./login-user.component.css']
})
export class LoginComponent implements OnInit {

  user: User = new User();

  constructor(private userService: UserService,
               private router: Router,
               private toast: ToastrService) {}
}
```

# login-user.component.ts (cont)

```
ngOnInit() {  
  let userSessionStr = sessionStorage.getItem("user_login");  
  if(userSessionStr){  
    this.user = JSON.parse(userSessionStr);  
    this.router.navigate(["list/user"]);  
  }  
}  
  
onSubmit(){  
  this.userService.retrieveByLogin(this.user.login)  
    .subscribe(  
    (res:User[])=>{  
      if(res.length>0){  
        if(res[0].password==this.user.password){  
          sessionStorage.setItem("user_login",JSON.stringify(res[0]));  
          this.router.navigate(["list/user"]);  
        }else{  
          console.log("Invalid password!");  
          this.toasty.error("Invalid password!")  
        }  
      }else{  
        console.log("User does not exist!");  
        this.toasty.error("User does not exist!")  
      }  
    }  
  );  
}
```

# login-user.component.html

```
<h2>Login</h2>
<hr>
<form (ngSubmit)="onSubmit()">

  <div class="form-group">
    <label for="login">Login</label>
    <input type="text" class="form-control" name="login" id="login"
      [(ngModel)]="user.login">
  </div>

  <div class="form-group">
    <label for="password">Password</label>
    <input type="password" class="form-control" name="password" id="password"
      [(ngModel)]="user.password">
  </div>

  <button type="submit" class="btn btn-primary">Enter</button>
  <a [routerLink]="['/register/user']" class="ml-2">Register</a>

</form>
```

# core.module.ts

```
import { NgModule } from '@angular/core';
import { FormsModule } from '@angular/forms';
import { CommonModule } from '@angular/common';
import { RegisterUserComponent } from './register-user/register-user.component';
import { ListUserComponent } from './list-user/list-user.component';
import { EditUserComponent } from './edit-user/edit-user.component';
import { LoginUserComponent } from './login-user/login-user.component';
import { BrowserAnimationsModule } from '@angular/platform-browser/animations';
import { ToastrModule } from 'ngx-toastr';
import { RouterModule } from '@angular/router';
```

```
@NgModule({
  declarations: [RegisterUserComponent, ListUserComponent, EditUserComponent, LoginUserComponent],
  exports: [RegisterUserComponent, ListUserComponent, LoginUserComponent],
  imports: [
    CommonModule,
    FormsModule,
    BrowserAnimationsModule,
    ToastrModule.forRoot(),
    RouterModule
  ]
})
export class CoreModule { }
```

# app.component.ts

```
import { User } from './models/User';
import { Component, OnInit } from '@angular/core';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent implements OnInit {

  user: User = null;

  constructor(){}

  ngOnInit(): void {
    let userSessionStorage = sessionStorage.getItem("user_login");
    if(sessionStorage){
      this.user = JSON.parse(userSessionStorage);
    }
  }

  logout(){
    sessionStorage.removeItem("user_login");
    this.user = null;
  }
}
```

# app.component.html

```
<nav class="navbar navbar-expand navbar-dark bg-dark" *ngIf="user">
  <div class="navbar-nav">
    <a class="nav-item nav-link" routerLink="register/user">Register</a>
    <a class="nav-item nav-link" routerLink="list/user">List</a>
  </div>

  <div class="collapse navbar-collapse justify-content-end">
    <span style="color: white; font-weight: bold;">Hello, {{user.firstName}}!</span>
    <button class="btn btn-secondary ml-2" (click)="logout()">Logout</button>
  </div>
</nav>

<div class="jumbotron">
  <div class="container">
    <div class="row">
      <div class="col-10 offset-1">
        <router-outlet></router-outlet>
      </div>
    </div>
  </div>
</div>
```

# app.module.ts

```
import { EditUserComponent } from './core/edit-user/edit-user.component';
import { ListUserComponent } from './core/list-user/list-user.component';
import { CoreModule } from './core/core.module';
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { HttpClientModule } from '@angular/common/http';
import { Routes, RouterModule } from '@angular/router';

import { AppComponent } from './app.component';
import { RegisterUserComponent } from './core/register-user/register-user.component';
import { LoginUserComponent } from './core/login-user/login-user.component';

const routes: Routes = [
  {path:"",component:LoginUserComponent},
  {path:'list/user',component:ListUserComponent},
  {path:'register/user',component:RegisterUserComponent},
  {path:'edit/user/:id',component>EditUserComponent}
];
```

# app.module.ts (cont)

```
@NgModule({  
  declarations: [  
    AppComponent  
  ],  
  imports: [  
    BrowserModule,  
    CoreModule,  
    HttpClientModule,  
    RouterModule.forRoot(routes)  
  ],  
  providers: [],  
  bootstrap: [AppComponent]  
})  
export class AppModule { }
```