import { Injectable } from '@angular/core';

import { Http, Headers } from '@angular/http';

import {tokenNotExpired} from 'angular2-jwt';

import 'rxjs/add/operator/map';

import { DeployService } from './deploy.service';

@Injectable()

export class AuthenticationService {

authToken: any;

user: any;

routeStart = this.deployService.getRouteStart();

constructor(private http:Http,

private deployService:DeployService) { }

registerUser(user) {

let headers = new Headers();

headers.append('Content-Type','application/json');

return this.http.post(this.routeStart + 'users/register', user, {headers: headers})

.map(res => res.json());

}

authenticateUser(user) {

let headers = new Headers();

headers.append('Content-Type','application/json');

return this.http.post(this.routeStart + 'users/authenticate', user, {headers: headers})

.map(res => res.json());

}

getProfile() {

let headers = new Headers();

this.loadToken();

headers.append('Authorization', this.authToken);

headers.append('Content-Type','application/json');

return this.http.get(this.routeStart + 'users/profile', {headers: headers})

.map(res => res.json());

}

storeUserData(data) {

localStorage.setItem('id\_token', data.token);

localStorage.setItem('user', JSON.stringify(data.user));

localStorage.setItem('projects', JSON.stringify(data.projects));

this.authToken = data.token;

}

loadToken() {

const token = localStorage.getItem('id\_token');

this.authToken = token;

}

loggedIn(){

return tokenNotExpired('id\_token');

}

logout() {

this.authToken = null;

this.user = null;

localStorage.clear();

}

searchUser(user) {

let headers = new Headers();

headers.append('Content-Type','application/json');

return this.http.post(this.routeStart + 'users/search', user, {headers: headers})

.map(res => res.json());

}

getUserData(user) {

let headers = new Headers();

headers.append('Content-Type','application/json');

return this.http.post(this.routeStart + 'users/getdata', user, {headers: headers})

.map(res => res.json());

}

}