import { Component, OnInit } from '@angular/core';

import { Router } from '@angular/router';

import { format } from 'date-fns';

import { FlashMessagesService } from 'angular2-flash-messages';

import \* as esLocale from 'date-fns/locale/es';

import { ProjectHandlingService } from '../../services/project-handling.service';

@Component({

selector: 'app-dashboard',

templateUrl: './dashboard.component.html',

styleUrls: ['./dashboard.component.css']

})

export class DashboardComponent implements OnInit {

projects: any;

newProjectName: String;

projectToRemove = {

\_id: "",

name: "",

updated: ""

};

userId: String;

constructor(

private flashMessagesService: FlashMessagesService,

private projectHandlingService: ProjectHandlingService,

private router: Router

) { }

ngOnInit() {

this.userId = JSON.parse(localStorage.getItem('user')).id;

this.projects = JSON.parse(localStorage.getItem('projects'));

this.formatDate();

}

onCreateProject() {

this.projectHandlingService.createProject(this.newProjectName).subscribe(data => {

if(data.success) {

this.flashMessagesService.show(data.msg, { cssClass: 'alert-success', timeout: 3000 });

data.project.imMaster = this.userId == data.project.masterId;

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

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

this.projects = data.projects;

this.formatDate();

this.router.navigate(['/principal']);

}

else {

this.flashMessagesService.show(data.msg, { cssClass: 'alert-info', timeout: 3000 });

}

});

}

onSelectProject(projectIndex) {

this.projectHandlingService.loadProject(projectIndex).subscribe(data => {

if(data.success) {

this.flashMessagesService.show(data.msg, { cssClass: 'alert-success', timeout: 3000 });

data.project.imMaster = this.userId == data.project.masterId;

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

this.router.navigate(['/principal']);

}

else {

this.flashMessagesService.show(data.msg, { cssClass: 'alert-info', timeout: 3000 });

}

});

}

onSelectProjectToRemove(project: any) {

this.projectToRemove = project;

}

onRemoveProject() {

const projectId = {\_id: this.projectToRemove.\_id}

this.projectHandlingService.deleteProject(projectId).subscribe(data => {

if(data.success) {

this.flashMessagesService.show(data.msg, { cssClass: 'alert-success', timeout: 3000 });

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

this.projects = data.projects;

this.formatDate();

}

else {

this.flashMessagesService.show(data.msg, { cssClass: 'alert-info', timeout: 3000 });

}

});

}

formatDate() {

for (let project of this.projects)

project.updated = format(project.updated, 'MMMM D [del] YYYY [a las] h:mm a', {locale: esLocale});

}

}