1. Build deploy

ng build

cd dist/

jar cvf portalweb.war .

1. Cli

npm i --legacy-peer-deps

ng add angular-datatables

1. Login
   1. Copy src/framework/auth to src/app/@theme/components/
   2. File auth.component.scss

Change @use '../../theme/styles/globals' as \*; to @use '@nebular/theme/styles/globals' as \*;

* 1. File app-routing.module.ts

Change import {NbAuthComponent,…} from '@nebular/auth'; to './@theme/components/auth/public\_api';

* 1. File core.module.ts

Change import { NbAuthModule,…} from '@nebular/auth'; to '../@theme/components/auth/public\_api';

const NB\_CORE\_PROVIDERS:

Remove NbDummyAuthStrategy

Replace by

NbPasswordAuthStrategy.setup({

name: 'email',

baseEndpoint:"http://localhost:4201/",

login: {

endpoint: 'auth',

method:"post",

redirect:{

success:"/pages"

},

requireValidToken:true

},

logout:{

endpoint:'signout',

method:'get',

requireValidToken:true

},

token: {

class: NbAuthJWTToken,

key:'token' //field name token of response

},

refreshToken: {

endpoint: '/refresh',

method: "post"

},

}),

* 1. File login.component.html
  2. File header.component.ts

constructor(

private authService: NbAuthService,

) {

this.authService.onTokenChange().subscribe((token: NbAuthJWTToken) => {

if (token.isValid()) {

console.log("token ", token);

// this.user = token.getPayload(); // here we receive a payload from the token and assigns it to our user variable

this.user = {

name: token.getPayload().sub,

roles: token.getPayload().roles,

picture: 'assets/images/nick.png',

token: token.getValue()

};

console.log("this.user ", this.user);

}

});

}

* 1. dasd

1. Logout
   1. File header.component.ts: add

userMenu = [

{ title: "Profile", icon: "people-outline" },

{ title: "Settings", icon: "settings-outline" },

{

title: "Log out",

icon: "log-out-outline",

link: "/auth/logout",

},

];

1. AuthGuard – auto redirect login if token null ([Nebular - Protecting application routes (akveo.github.io)](https://akveo.github.io/nebular/docs/auth/protecting-application-routes#related-articles))
   1. Add new file: auth-guard.service.ts

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

import { NbAuthService } from '../public\_api';

@Injectable()

export class AuthGuard implements CanActivate {

constructor(private authService: NbAuthService, private router: Router) {

}

canActivate() {

return this.authService.isAuthenticated()

.pipe(

tap(authenticated => {

if (!authenticated) {

this.router.navigate(['auth/login']);

}

}),

);

}

}

* 1. Register the service: file app.modules.ts

@NgModule({

imports: [

],

declarations: [

],

exports: [

],

providers: [

AuthGuard

],

})

* 1. Add guard to routes: file app-routing.modules.ts

{

path: 'pages',

canActivate: [AuthGuard],

loadChildren: () => import('./pages/pages.module')

.then(m => m.PagesModule),

},

1. Interceptor – add token per request api
   1. File app.module.ts add

@NgModule({

declarations: [AppComponent],

imports: [

],

bootstrap: [AppComponent],

providers: [

AuthGuard,

{ provide: APP\_BASE\_HREF, useValue: '/' },

{ provide: NB\_AUTH\_TOKEN\_INTERCEPTOR\_FILTER, useValue: function () { return false; }, },

{ provide: HTTP\_INTERCEPTORS, useClass: NbAuthJWTInterceptor, multi: true },

],

})

1. Demo page

app/pages/pages-menu.ts

app/pages/ pages-routing.module.ts

ng g c pages/demo

ng g s pages/demo

ng g m pages/demo

ng g c pages/demo/demo-table

ng g c pages/demo/user-table

ng g c pages/demo/sasmasterdata-table

ng g c pages/demo/create-or-edit-user-modal

ng g class pages/demo/user

ng g class pages/demo/role

ng g class pages/demo/department

ng update @angular/core@14 @angular/cli@14 --allow-dirty --force

ng update @angular/core@15 @angular/cli@15 --allow-dirty –force

npm uninstall @angular-devkit/build-angular

npm install --save-dev @angular-devkit/build-angular

Add DemoModule in imports: PagesModule  
@NgModule({

imports: [

DemoModule

],

declarations: [

],

})

Add DemoComponent, DemoTableComponent, UserTableComponent, SasmasterdataTableComponent in declarations: DemoModule

@NgModule({

imports: [

Ng2SmartTableModule,

],

declarations: [

DemoComponent,

DemoTableComponent,

UserTableComponent,

SasmasterdataTableComponent

]

})

* 1. asd

1. Skip generate test file
   1. File angular.json

"schematics": {

"@schematics/angular:component": {

"prefix": "ngx",

"style": "scss",

"skipTests": true

},

"@schematics/angular:directive": {

"prefix": "ngx",

"skipTests": true

},

"@schematics/angular:class": {

"skipTests": true

},

"@schematics/angular:directive": {

"skipTests": true

},

"@schematics/angular:pipe": {

"skipTests": true

},

"@schematics/angular:service": {

"skipTests": true

},

}

1. Fix angular datatable

[angularjs - Can't bind to 'dtOptions' since it isn't a known property of 'table'. - Stack Overflow](https://stackoverflow.com/questions/46157303/cant-bind-to-dtoptions-since-it-isnt-a-known-property-of-table)

1. Get token from localStorage

const token = JSON.parse(localStorage.getItem('auth\_app\_token')).value;

1. Datatable filter multi column on top

tfoot {

display: table-header-group;

}

1. Role page

Lấy role từ db

File deader.component: this.aclService.setAccessControl(permissions);

1. Role page

app/pages/pages-menu.ts

app/pages/pages-routing.module.ts

ng g c pages/administration

ng g m pages/administration –routing

ng g c pages/administration/role

ng g c pages/administration/role/role-table

ng g c pages/administration/role/create-or-edit-role-modal

ng g c pages/administration/user

ng g c pages/administration/user/user-table

ng g c pages/administration/user/create-or-edit-user-modal

ng g s shared/services/user/user

ng g class shared/services/user/dto/userDto

ng g s shared/services/role/role

ng g class shared/services/role/dto/roleDto

ng g c pages/administration/audit-log

ng g c pages/administration/audit-log/audit-log-table

ng g c pages/administration/audit-log/audit-log-modal

ng g s shared/services/audit-log/audit-log

ng g class shared/services/audit-log/dto/audit-log-dto

ng g c pages/home

1. Thêm roles vào token localStorage: \app\@theme\components\auth\services\token\token-parceler.ts

Thêm vào trong function wrap()

pages.component.ts

1. Config global Modal

Thêm trong file app.module.ts

NbDialogModule.forRoot({

hasBackdrop: true,

closeOnBackdropClick: false,

closeOnEsc: true,

hasScroll: true

}),

1. Refresh token

File auth.service.ts

Func refreshToken

Thay đổi data input của api refreshToken

1. Thay đổi sang NbAuthOAuth2JWTToken (access\_token, refresh\_token)

File core.module

token: {

class: NbAuthOAuth2JWTToken,

getter: function (method: any, response: any) {

return {

access\_token: response.body.token,

refresh\_token: response.body.refreshToken,

};

},

},

File token-parceler

File auth-gurad.service

File pages.component

File header.component

1. dfgdfg