



EXPLORER

OPEN EDITORS

UNTITLED (WORKSPACE)

projects

src

app

admin

benefits

common

animation

TS fade-animation.ts

TS route-animation.ts

consts

directives

TS visa-has-permission.directive.ts

TS visa-is-permitted.directive.spec.ts

errors

TS error-handler.ts

TS errors.module.ts

TS redirect.interceptor.spec.ts

TS redirect.interceptor.ts

TS server-errors.interceptor.ts

guards

TS auth.guard.spec.ts

TS auth.guard.ts

TS dev.guard.spec.ts

TS dev.guard.ts

TS roles-permissions.guard.spec.ts

TS roles-permissions.guard.ts

injection-tokens

pipes

services

loading-block.component.html

loading-block.component.scss

TS loading-block.component.ts

permission-control.component.html

config

core

data

error

files

guide

home

TS files-service.ts

TS roles-permissions.guard.ts

```
1 import { Injectable } from '@angular/core';
2 import { ActivatedRouteSnapshot, CanActivate, Router, RouterStateSnapshot } from '@angular/router';
3 import { Observable } from 'rxjs';
4 import { UserContextUtils } from '../../../../../shared/utils/user-context-utils';
5 import { NotificationDuration, NotificationsService, NotificationType } from '../../../../../shared/notifications/notifications.service';
6 import { MessagesService } from '../services/messages.service';
7 // import { MESSAGES } from '../consts/messages';
8
9 @Injectable()
10 export class RolesPermissionsGuard implements CanActivate {
11   constructor(
12     private notificationsService: NotificationsService,
13     private userContextUtils: UserContextUtils,
14     private messagesService: MessagesService
15   ) {}
16
17   public canActivate(next: ActivatedRouteSnapshot, state: RouterStateSnapshot): Observable<boolean> | Promise<boolean> | boolean {
18     if (next.data.permissions && this.userContextUtils.userContext.scopes.some(permission => next.data.permissions.includes(permission))) {
19       return true;
20     } else {
21       this.notificationsService.show(this.messagesService.MESSAGES.REQUEST_NOT_ALLOWED, NotificationType.Error, NotificationDuration.Short);
22       return false;
23     }
24   }
25 }
26
```



OUTLINE



EXPLORER

OPEN EDITORS

UNTITLED (WORKSPACE)

projects

src

app

admin

benefits

common

animation

TS fade-animation.ts

TS route-animation.ts

consts

directives

TS visa-has-permission.directive.ts

TS visa-is-permitted.directive.spec.ts

errors

TS error-handler.ts

TS errors.module.ts

TS redirect.interceptor.spec.ts

TS redirect.interceptor.ts

TS server-errors.interceptor.ts

guards

TS auth.guard.spec.ts

TS auth.guard.ts

TS dev.guard.spec.ts

TS dev.guard.ts

TS roles-permissions.guard.spec.ts

TS roles-permissions.guard.ts

injection-tokens

pipes

TS files-service.ts

TS auth.guard.ts

```
1 import { Injectable } from '@angular/core';
2 import { Router, CanActivate } from '@angular/router';
3
4 @Injectable()
5 export class AuthGuard implements CanActivate {
6   constructor(private router: Router) {}
7
8   public canActivate() {
9     if (sessionStorage.getItem('currentUser')) {
10       // Logged in so return true
11       return true;
12     }
13
14     // not logged in so redirect to login page
15     this.router.navigate(['/sessionExpired']);
16     return false;
17   }
18 }
19 }
```

EXPLORER

OPEN EDITORS

UNTITLED (WORKSPACE)

AngularVDBP

.sonarlint

vscode

coverage

dist

e2e

node_modules

projects

src

app

admin

benefits

common

animation

TS fade-animation.ts

TS route-animation.ts

consts

directives

TS visa-has-permission.directive.ts

TS visa-is-permitted.directive.specs.ts

errors

guards

injection-tokens

pipes

services

loading-block.component.html

loading-block.component.scss

TS loading-block.components.ts

permission-control.component.html

config

core

data

error

files

guide

home

login-page

logout

TS files-service.ts TS visa-has-permission.directive.ts

```
1 import { Directive, ElementRef, Input, OnInit, TemplateRef, ViewContainerRef } from '@angular/core';
2 import { UserContextUtils } from '../../../../../shared/utils/user-context-utils';
3
4 @Directive({
5   selector: '[visaHasPermission]',
6 })
7 export class VisaHasPermissionDirective {
8   constructor(
9     private element: ElementRef,
10    private templateRef: TemplateRef<any>,
11    private viewContainer: ViewContainerRef,
12    private userContextUtils: UserContextUtils
13  ) {}
14
15 @Input()
16 set visaHasPermission(permissions) {
17   if (permissions && !this.isPermitted(permissions)) {
18     this.viewContainer.clear();
19   } else {
20     this.viewContainer.createEmbeddedView(this.templateRef);
21   }
22 }
23
24 private isPermitted(permissions: string[]) {
25   const userScope = this.userContextUtils.userContext.scopes;
26   if (userScope && userScope.some(permission => permissions.includes(permission))) {
27     return true;
28   } /* else {
29   console.log('NOT PERMITTED >>', permissions);
30   }*/
31   return false;
32 }
33 }
34 }
```



EXPLORER

OPEN EDITORS

UNTITLED (WORKSPACE)

AngularVDBP

.sonarlint

.vscode

coverage

dist

e2e

node_modules

projects

src

app

admin

components

containers

benefit-types

benefit-types.component.html

benefit-types.component.scss

TS benefit-types.component.ts

issuers

new-benefit-type

organizations

products

spend-tiers

spend-tiers-eligibility

services

TS admin.module.ts

TS admin.routing.ts

benefits

common

animation

TS fade-animation.ts

TS route-animation.ts

consts

directives

TS visa-has-permission.directive.ts

TS visa-is-permitted.directive.spec.ts

TS files-service.ts

```
1  <visa-container>
2    <div class="container">
3      <div class="row">
4        <div class="col-3">
5          | <h1 i18n>Benefit Types</h1>
6        </div>
7
8        <div class="col-9">
9          <visa-buttons>
10         <button *visaHasPermission="[PERMISSIONS.MANAGE_VDBP_BENEFIT_TYPE]" class="button button--empty" type="button" [routerLink]=[['/n
11           | + New Benefit Type
12         </button>
13
14         <button class="button" type="button" [visaOverviewPanelTriggerFor]="searchPanel" i18n>
15           | {{open ? 'Done' : 'Search'}}
16         </button>
17         <visa-overview-panel #searchPanel="visaOverviewPanel" (toggled)="actionOnOverviewPanelToggled($event)" container="body">
18           <visa-benefit-type-search
19             #benefitTypeSearchPanelForm
20             [panel]="searchPanel"
21             (search)="onSearch($event)"
22             title="Search Benefit Types"
23             [benefitTypesCodes]="benefitTypesCodes"
24             i18n-title
25           ></visa-benefit-type-search>
26         </visa-overview-panel>
27
28         </visa-buttons>
29       </div>
30     </div>
31
32     <div class="row table-row">
33       <div class="col-12">
34         <visa-table *ngIf="items?.length else noRecord" [headers]="headers" [items]="items" [itemsPerPage]="10" (change)="onPageItemsChang
35         <tr *ngFor="let item of tablePageItems; let index = index" class="table-list-tr" #row (click)="editBenefitType(item)">
36           <td visaTableCell="benefitTypeCode" class="benefit-type-code">{{item.benefitTypeCode}}</td>
37           <td visaTableCell="benefitTypeName" class="benefit-type-code">{{item.benefitTypeName}}</td>
38           <td visaTableCell="benefitTypeDescription" class="benefit-type-code">{{item.benefitTypeDesc}}</td>
39         </tr>
40       </visa-table>
41       <ng-template #noRecord i18n>No Record Available</ng-template>
42     </div>
43   </div>
44 </div>
```



EXPLORER

- ▶ OPEN EDITORS
- ◀ UNTITLED (WORKSPACE)
 - app
 - admin
 - ▶ components
 - ▶ containers
 - ▶ services
 - TS admin.module.ts
 - TS admin.routing.ts
 - ▶ benefits
 - ▶ common
 - ▶ animation
 - TS fade-animation.ts
 - TS route-animation.ts
 - ▶ consts
 - ▶ directives
 - TS visa-has-permission.directive.ts
 - TS visa-is-permitted.directive.spec.ts
 - ▶ errors
 - ▶ guards
 - ▶ injection-tokens
 - ▶ pipes
 - TS card-mask.pipe.ts
 - TS categories-list.pipe.ts
 - TS invert-color.pipe.ts
 - TS range.pipe.spec.ts
 - TS range.pipe.ts
 - TS replace-variable.pipe.spec.ts
 - TS replace-variable.pipe.ts
 - ▶ services
 - ▷ loading-block.component.html
 - Ρ loading-block.component.scss
 - TS loading-block.component.ts
 - ▷ permission-control.component.html
 - ▶ config
 - ▶ core
 - ▶ data

TS files-service.ts TS card-mask.pipe.ts x

```
1 import { Pipe, PipeTransform } from '@angular/core';
2
3 @Pipe({
4   name: 'mask',
5 })
6 export class CardMask implements PipeTransform {
7   public transform(card: string): string {
8     if (card) {
9       const first = card.substring(0, 6);
10      const last = card.substring(12, card.length);
11      return first + 'X'.repeat(6) + last;
12    }
13  }
14 }
15
```

```
| mask
| Aa Ab! *
| 21 <div class="secondary-navigation dark">
| 22   <ul class="no-border no-margin">
| 23     <li *ngFor="let secondaryNavigationSection of secondaryNavigationSections" [ngClass]={'active':secondaryNavigationSe
| 24       <h4 i18n>{{secondaryNavigationSection.label}}</h4>
| 25     </ul>
| 26   </div>
| 27
| 28
| 29   <div class="searched-data-info">
| 30     <h4 class="messages" i18n *ngIf="cardAccount.caqResponseHeader != null">{{cardAccount.caqResponseHeader.responseDescript
| 31
| 32
| 33     <div class="m-t-30">
| 34       <h4 i18n>Card Account</h4>
| 35     </div>
| 36
| 37     <visa-form-input
| 38       class="visa-form-field visa-form-field--readonly"
| 39       type="text"
| 40       readonly="true"
| 41       [doNotShowValidationState]="true"
| 42       label="Card Account Number"
| 43       i18n-label
| 44       [hideLabel]="true"
| 45       [ngModel]="requestData.cardAccountNumber | mask">
| 46     </visa-form-input>
| 47     <visa-form-input
| 48       class="visa-form-field visa-form-field--readonly"
| 49       type="text"
| 50       readonly="true"
| 51       [doNotShowValidationState]="true"
| 52       label="As of date"
| 53       i18n-label
| 54       [ngModel]="requestData.asOfDate | date: 'MM/dd/yyyy'">
| 55     </visa-form-input>
| 56   </div>
| 57   </visa-sticky-box>
| 58 </div>
| 59 </div>
| 60 </div>
| 61
| 62 <div class="col-9" *ngIf="secondaryNavigationSections[0].active">
| 63   <visa-card-account-query-general-information [cardAccount]='cardAccount'></visa-card-account-query-general-information>
| 64 </div>
| 65 <div class="col-9" *ngIf="secondaryNavigationSections[1].active">
| 66   <visa-card-account-query-benefits [allBenefits]='cardAccount.benefits'></visa-card-account-query-benefits>
```

SEARCH

mask

Replace

1 result in 1 file

< card-account-query-overview.component.html src\app\more... 1
requestData.cardAccountNumber | mask >

Aa Abi *



TS files-service.ts

zo

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

40

41

42

43

44

45

46

47

48

49

50

51

52

53

54

55

56

57

58

59

60

61

<> card-account-query-overview.component.html x

<V15D+SLCKY-DUX>

<div class="secondary-navigation dark">
| <ul class="no-border no-margin">
| | <li *ngFor="let secondaryNavigationSection of secondaryNavigationSections" [ngClass]="{{'active':secondaryNavigationSection.active}}>
| | | <h4>{{secondaryNavigationSection.label}}</h4>
| |

</div><div class="searched-data-info">
| <h4 class="messages" i18n *ngIf="cardAccount.caqResponseHeader != null">{{cardAccount.caqResponseHeader.responseDescription}}<div class="m-t-30">
| <h4 i18n>Card Account</h4>
</div><visa-form-input
| class="visa-form-field visa-form-field--readonly"
| type="text"
| readonly="true"
| [doNotShowValidationState]="true"
| label="Card Account Number"
| i18n-label
| [hideLabel]="true"
| [ngModel]="requestData.cardAccountNumber | mask">
</visa-form-input><visa-form-input
| class="visa-form-field visa-form-field--readonly"
| type="text"
| readonly="true"
| [doNotShowValidationState]="true"
| label="As of date"
| i18n-label
| [ngModel]="requestData.asOfDate | date: 'MM/dd/yyyy'">
</visa-form-input></div>
</visa-sticky-box>
</div>
</div>
</div>

" : 1, "o": 2, "c": "secondaryNavigationSections[0].active"



EXPLORER

OPEN EDITORS

UNTITLED (WORKSPACE)

animation

TS fade-animation.ts

TS route-animation.ts

consts

directives

TS visa-has-permission.directive.ts

TS visa-is-permitted.directive.spec.ts

errors

guards

injection-tokens

pipes

TS card-mask.pipe.ts

TS categories-list.pipe.ts

TS invert-color.pipe.ts

TS range.pipe.spec.ts

TS range.pipe.ts

TS replace-variable.pipe.spec.ts

TS replace-variable.pipe.ts

services

TS ajax-interceptor.service.spec.ts

TS ajax-interceptor.service.ts

TS authentication.service.spec.ts

TS authentication.service.ts

TS benefits.service.ts

TS consts.service.ts

TS http-mock.service.ts

TS idle-user.service.ts

TS loading-block.service.ts

TS messages.service.ts

TS metadata.service.ts

TS request-count.service.ts

TS ajax-interceptor.service.ts

```
1 import { HttpEvent, HttpHandler, HttpInterceptor, HttpRequest } from '@angular/common/http';
2 import { Injectable } from '@angular/core';
3 import { Observable } from 'rxjs';
4
5 @Injectable()
6 export class AjaxInterceptorService implements HttpInterceptor {
7   constructor() {}
8
9   public intercept(req: HttpRequest<any>, next: HttpHandler): Observable<HttpEvent<any>> {
10    let headers = req.headers;
11
12    const token: string = sessionStorage.getItem('currentUser') ? JSON.parse(sessionStorage.getItem('currentUser')).token : '';
13    if (token) {
14      headers = headers.set('Authorization', 'Bearer ' + token);
15    }
16
17    if (!req.headers.has('Content-Type') && !(req.body instanceof FormData)) {
18      headers = headers.set('Content-Type', 'application/json');
19    }
20
21    if (!req.headers.has('Accept')) {
22      headers = headers.set('Accept', 'application/json');
23    }
24
25    if (!req.headers.has('Access-Control-Allow-Origin')) {
26      headers = headers.set('Access-Control-Allow-Origin', '*');
27    }
28
29    req = req.clone({ headers });
30    return next.handle(req);
31  }
32}
33
```



SEARCH

AjaxInterceptorService

Aa Abi *

Replace

16 results in 7 files

- ▲ TS admin.module.ts src\app\admin
 - import { AjaxInterceptorService } from 'app/common/services/ajax-i...
useClass: AjaxInterceptorService,
- ▲ TS app.module.ts src\app
 - import { AjaxInterceptorService } from './common/services/ajax-inte...
useClass: AjaxInterceptorService,
- ▲ TS benefits.module.ts src\app\benefits
 - import { AjaxInterceptorService } from './common/services/ajax-inte...
useClass: AjaxInterceptorService,
- ▲ TS ajax-interceptor.service.spec.ts src\app\common\services
 - import { AjaxInterceptorService } from './ajax-interceptor.service';
describe('AjaxInterceptorService', () => {
 providers: [AjaxInterceptorService],
 inject([AjaxInterceptorService], (service: AjaxInterceptorService) => {
 AjaxInterceptorService, (service: AjaxInterceptorService) => {
- ▲ TS ajax-interceptor.service.ts src\app\common\services
 - export class AjaxInterceptorService implements HttpInterceptor {
- ▲ TS files.module.ts src\app\files
 - import { AjaxInterceptorService } from 'app/common/services/ajax-i...
useClass: AjaxInterceptorService,
- ▲ TS more.module.ts src\app\more
 - import { AjaxInterceptorService } from 'app/common/services/ajax-i...
useClass: AjaxInterceptorService,

TS app.module.ts x

```
providers: [  
  // ...  
  // !!! IMPORTANT !!!  
  // ...  
  // Multilingual - The configuration needs to be in shared module to avoid  
  // StaticInjectorError(Platform: core)[I18n -> InjectionToken TranslationsFormat]  
  // for 'ngx-translate/i18n-polyfill' I18n service  
  // { provide: LOCALE_ID, useValue: 'en' },  
  I18n,  
  {  
    provide: HTTP_INTERCEPTORS,  
    useClass: AjaxInterceptorService,  
    multi: true,  
  },  
  {  
    provide: HTTP_INTERCEPTORS,  
    useClass: RequestCountService,  
    multi: true,  
  },  
  {  
    provide: LocationStrategy,  
    useClass: HashLocationStrategy,  
  },  
  AppConfig,  
  {  
    provide: APP_INITIALIZER,  
    useFactory: initializeApp,  
    deps: [AppConfig],  
    multi: true,  
  },  
  LoadingBlockService,  
  AuthenticationService,  
  UsersService,  
  UrlService,  
  HttpMock,  
  AuthGuard,  
  BenefitService,  
  BenefitAttachmentService,  
  RolesPermissionsGuard,  
  BenefitTypeService,  
  MessagesService,  
  ConstsService,  
  MetadataService,  
],  
bootstrap: [AppComponent],
```