

UploadFileComponent

August 17, 2026

UploadFileComponent

Kind: Class

Source: Frontend/src/app/components/files/upload-file/upload-file.component.ts (line 21)

Part of: Frontend

UploadFileComponent manages file-upload UI state and interactions in the Angular frontend. It handles initialization, window resizing, upload completion, file exclusion, text selection, and synchronization with the upload popup.

Implements: OnInit

Methods

Method	Signature	Returns
ngOnInit	ngOnInit()	void
resizeWindow	resizeWindow()	void
getNumberFormatted	getNumberFormatted(x: undefined)	string
isMeDeveloper	isMeDeveloper()	boolean
selectText	selectText(e: undefined)	void
stopPropagation	stopPropagation(e: undefined)	void
traceTypesListName	traceTypesListName(value: number)	string
syncWithUploadPopup	syncWithUploadPopup()	void
onUploaded	onUploaded()	void
excludeFile	excludeFile(file: undefined, index: number)	void
GetTagsListByType	GetTagsListByType(Type: number, supplierId: number)	void
GetCategoriesList	GetCategoriesList()	void

Method	Signature	Returns
getUsersPermission	getUsersPermission()	void
getFilesList	getFilesList(parentId: string)	void
trace	trace(data: any)	void
getFoldersList	getFoldersList(parentId: string, levelIndex: number)	void
extractFilesExtension	extractFilesExtension(file: OneDriveFilesList)	string
onChooseTagType	onChooseTagType(e: undefined)	void
saveNewFile	saveNewFile()	void
selectFile	selectFile(file: OneDriveFilesList)	void
closePopup	closePopup()	void
LinkTag	LinkTag(doLinkDirectly: boolean)	void
DeleteTag	DeleteTag(idx: number)	void
addComment	addComment()	void
DeleteComment	DeleteComment(idx: number)	void
ngOnDestroy	ngOnDestroy()	void
changeTabs	changeTabs(targetTab: undefined)	void
selectItem	selectItem(e: any)	void
onInitialized	onInitialized(e: undefined)	void

Properties

Property	Type
treeView	DxTreeViewComponent
uploadValidation	DxValidationGroupComponent
TagReferenceToTypeID	number
TagReferenceToID	number
ParentFileID	number
tempSelectedDataFile	FileData
SelectedDataFiles	FileData[]

Property	Type
tempShareFile	OneDriveShareFiles
ShareFiles	OneDriveShareFiles[]
usersPermission	DriveUser[]
SelectedList	OneDriveFilesList[]
FoldersList	OneDriveFilesList[]
fileNameAndExtension	FileNameAndExtension[]
currentTab	string
newTag	UploadNewFile
UploadedFiles	File[]
uploadPopup	boolean
CompanyBranchID	number
selectedTagReferenceToTypeID	number
selectedTagReferenceToID	number
selectedTagRegistrationReferenceToSupplierID	number
displayNamePattern	any
FileRelatedToTypes	any
modalOptions	any
TypesList	any[]
TagLists	GeneralListView[][]
TagListDataSources	DataSource[]
categoriesListDataSources	any[]
Comment	string
subscription	Subscription
DatagridHeight	number
TotalFilesCount	number
TotalFoldersCount	number
updateTopContent	any
updateBottomContent	any

Property	Type
updateContentTimer	any
updateContent	any

Where it refuses work

- UploadFileComponent stops the work with an early return when `findIndex > -1` .
- UploadFileComponent stops the work with an early return when `!this.selectedTagReferenceToID || this.selectedTagReferenceToID == 0` .
- UploadFileComponent stops the work with an early return when `this.tempSelectedDataFile && this.tempSelectedDataFile.FileTags && this.tempSelectedDataF...` .
- UploadFileComponent stops the work with an early return when `!(this.Comment.trim() && this.Comment.trim().length > 0)` .

Diagram

```
graph LR
  User[User interaction] --> Component[UploadFileComponent]
  Component --> Init[ngOnInit]
  Component --> Resize[resizeWindow]
  Component --> Popup[syncWithUploadPopup]
  Component --> Uploaded[onUploaded]
  Component --> Exclude[excludeFile]
  Component --> Display[getNumberFormatted]
  Component --> Types[traceTypesListName]
  Component --> Access[isMeDeveloper]
```

Usage

```
import { Component, ViewChild } from '@angular/core';
import { UploadFileComponent } from './components/files/upload-file/upload-file.component';

@Component({
  selector: 'app-upload-page',
  template: `
    <app-upload-file #uploadFile></app-upload-file>

    <button type="button" (click)="openUploadPopup()">
      Open upload
    </button>
  `
})
```

```

})
export class UploadPageComponent {
  @ViewChild('uploadFile')
  uploadFileComponent!: UploadFileComponent;

  openUploadPopup(): void {
    this.uploadFileComponent.syncWithUploadPopup();
  }

  handleUploadComplete(): void {
    this.uploadFileComponent.onUploaded();
  }
}

```

AI Coding Instructions

- Keep upload-related UI state changes inside `UploadFileComponent` methods rather than duplicating them in parent components.
- Call `syncWithUploadPopup()` when external UI actions need to align with the upload popup state.
- Preserve event handling through `stopPropagation()` when nested controls must not trigger parent click handlers.
- Use `onUploaded()` after an upload completes so component state is updated through the existing flow.
- Check `isMeDeveloper()` before rendering or enabling developer-specific behavior.

Used by

1 reference from 1 file. Each is a place in this repository where the symbol is actually used — go read one rather than trusting an example.

Imported by (1)

- `TranslateHandler` — `Frontend/src/app/components/shared/shared.module.ts` :1