



Découverte de Compodoc

OUTIL DE DOCUMENTATION POUR APPLICATION ANGULAR

Willy Marchais | TP ALQ | 06/04/2020

Enseignant : David Hill

Présentation de Compodoc

Compodoc est un outil de documentation automatique pour les applications Angular. Il permet de générer une documentation détaillée de l'application. Le but de cette documentation est de pouvoir lire et comprendre les différents modules du code Angular simplement et visuellement.

Voici un exemple de projet documenté avec Compodoc :

The screenshot shows the 'Overview' page of a Compodoc-generated documentation site. On the left is a sidebar with a search bar and a list of navigation items: Getting started, Overview (active), README, Dependencies, Modules, Components, Injectables, Interceptors, Guards, Interfaces, Miscellaneous, and Documentation coverage. The main content area is titled 'Overview' and features a large, colorful dependency graph at the top. Below the graph are several statistics cards: 16 Modules, 18 Components, 1 Directive, 7 Injectables, 1 Pipe, 6 Guards, and 6 Interfaces. At the bottom left, it says 'Documentation generated using compodoc' with the logo.

This screenshot shows the 'Modules / ArticleModule' page. The left sidebar lists a tree of modules: ArticleModule (selected), ArticleRoutingModule, AuthModule, AuthRoutingModule, CoreModule, EditorModule, EditorRoutingModule, HomeModule, HomeRoutingModule, ProfileModule, ProfileRoutingModule, SettingsModule, SettingsRoutingModule, and SharedModule. The main area displays a diagram of the ArticleModule's dependencies, showing it importing SharedModule, ArticleRoutingModule, MarkdownPipe, ArticleComponent, and ArticleCommentComponent. A legend below the diagram explains the color coding: yellow for Declarations, teal for Module, blue for Bootstrap, orange for Providers, and red for Exports. At the bottom, there is a 'Source' tab showing the corresponding TypeScript code for ArticleModule.ts.

```
1 import { ModuleWithProviders, NgModule } from '@angular/core';
2 import { RouterModule } from '@angular/router';
3
4 import { ArticleComponent } from './article.component';
5 import { ArticleCommentComponent } from './article-comment.component';
6 import { ArticleResolver } from './article-resolver.service';
7 import { MarkdownPipe } from './markdown.pipe';
8 import { SharedModule } from '../shared';
9 import { ArticleRoutingModule } from './article-routing.module';
10
11 @NgModule({
12   imports: [
13     SharedModule,
14     ArticleRoutingModule
15   ],
16   declarations: [
17     ArticleComponent,
18     ArticleCommentComponent,
19     MarkdownPipe
20   ],
21   providers: [
22     ArticleResolver
23   ]
24 })
25 export class ArticleModule {}
```

Installation

Quelques pré requis avant d'installer Compodoc.

- **Node.js**
Les dernières versions de Node fonctionnent très bien avec Compodoc. Pour vérifier votre version utilisez la commande suivante :
`run node -v`
Pour télécharger Node.js : nodejs.org
- **Angular-CLI**, pour installer angular cli écrivez la commande suivante :
`npm install -g @angular/cli`

Avant d'installer Compodoc il faut un projet Angular, pour cela si vous n'en avez pas, vous pouvez télécharger, cloner un exemple de projet sur git avec le lien suivant :

<https://github.com/gothinkster/angular-realworld-example-app.git>

Installer Compodoc globalement avec **npm** (nécessite d'avoir **npm**, on peut vérifier cela avec la commande `npm -v`) :

```
npm install -g @compodoc/compodoc
```

Ensuite il suffit d'écrire la ligne ci-dessous du script dans le fichier `package.json` du projet Angular :

```
"scripts": { "compodoc": "npx compodoc -p src/tsconfig.app.json"}
```

Enfin on peut lancer la commande suivante pour générer la documentation :

```
npm run compodoc
```

Une fois la commande exécutée un message du terminal indique la génération du dossier Documentation et peut indiquer une URL.

Documentation generated in ./documentation/ in 4.677 seconds using gitbook theme

Fonctionnalités de Compodoc

Après avoir générer la documentation, on peut apercevoir dans notre dossier du projet Angular un répertoire nommé Documentation. Ce dossier est composé de fichiers html, on peut ouvrir le fichier index.html pour arriver sur notre projet documenté. Compodoc comprend directement que l'application est une application Angular, c'est-à-dire qu'il détermine les concepts comme Components, Pipe ou Directives. On peut voir cela avec le screen ci-dessous :

ang2-conduit documentation

Type to search

🏠 Getting started

📊 Overview

📖 README

⋮ Dependencies

📁 Modules

⚙️ Components

⬇️ Injectables

↔️ Interceptors


🔒 Guards

🕒 Interfaces

📦 Miscellaneous

📊 Documentation coverage

Documentation generated using


compodoc

Components / ArticleListComponent

InfoSourceTemplateStylesDOM Tree

File

src/app/shared/article-helpers/article-list.component.ts

Metadata

selector	app-article-list
styleUrls	article-list.comp
templateUrl	./article-list.co

Index

Properties

currentPage	que
loading	res

Methods

runQuery	set
----------	-----

Inputs

config	lim
--------	-----

Accessors

On peut consulter les propriétés, les constructeurs, les méthodes et attributs du Component sélectionné. Il est également possible de regarder le code directement dans l'onglet **Source**, le template HTML dans l'onglet **Template**, la feuille de style scss, css dans l'onglet **Styles** et la structure sous forme d'arborescence dans l'onglet **DOM**.

Il est également possible de consulter les modules avec cette documentation puisqu'Angular est réalisé à partir de modules :

Modules / ArticleModule

```
graph TD; subgraph LeftBox [ ]; SharedModule[SharedModule]; ArticleRoutingModule[ArticleRoutingModule]; end; subgraph RightBox [ ]; MarkdownPipe[MarkdownPipe]; ArticleComponent[ArticleComponent]; ArticleCommentComponent[ArticleCommentComponent]; end; ArticleModule[ArticleModule]; SharedModule --> ArticleModule; ArticleRoutingModule --> ArticleModule; ArticleModule --> MarkdownPipe; ArticleModule --> ArticleComponent; ArticleModule --> ArticleCommentComponent;
```

Legend

Declarations Module Bootstrap Providers Exports

Zoom in Reset Zoom out

Info Source

File

src/app/article/article.module.ts

Declarations ⓘ

ArticleCommentComponent

ArticleComponent

MarkdownPipe

Imports ⓘ

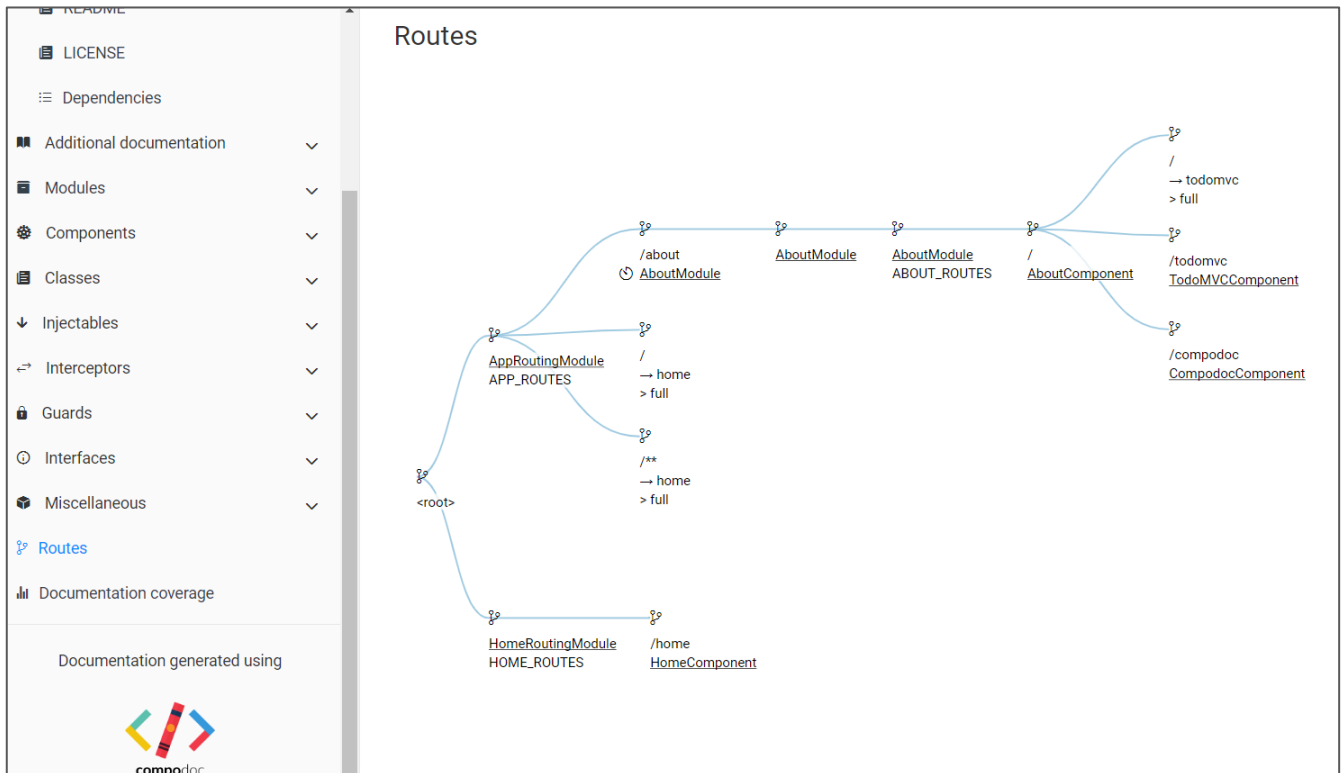
ArticleRoutingModule

SharedModule

Une démonstration d'une documentation en ligne est disponible à l'adresse suivante si vous n'avez pas de projet Angular :

<https://compodoc.github.io/compodoc-demo-todomvc-angular/>

Une autre fonctionnalité intéressante est la fonctionnalité **Routes** qui permet d'afficher sous forme de diagramme le chemin du projet :



Le type de commentaire accepté par la documentation est uniquement celui-ci :

```
/**
 * Exemple d'information à insérer ici !
 */
```

Résultat avec et sans commentaire :

Injectables / CommentsService

Info

Source

File

src/app/core/services/comments.service.ts

Index

Methods

add

destroy

Constructor

```
constructor(apiService: ApiService)
```

Defined in src/app/core/services/comments.service.ts:10

Injectables / CommentsService

Info

Source

File

src/app/core/services/comments.service.ts

Description

Exemple d'information à insérer ici !

Index

Methods

add

destroy

Constructor

D'autres paramètres pour les types de commentaires :

```
/**
 * @param slug ici un truc
 * @param payload et là un autre truc
 * @returns void
 */
```

add

`add(slug, payload)`

Defined in [src/app/core/services/comments.service.ts:26](#)

Parameters :

Name	Optional	Description
slug	No	ici un truc
payload	No	et là un autre truc

Returns : [Observable<Comment>](#)

void

Il existe pleins d'autres fonctionnalités intéressantes sur Compodoc. Pour en savoir plus il suffit de consulter le site Compodoc notamment le getting started : <https://compodoc.app/guides/getting-started.html>