

Documentation de projets

Pourquoi documenter ses projets

La documentation communautaire est nécessaire pour permettre à la communauté de comprendre le fonctionnement de votre programme/ressource sans passer par le code source, elle peut être tout aussi nécessaire pour des projets en interne, même s'il n'y a qu'un seul développeur présent.

Documenter un projet permet de s'assurer que son fonctionnement restera clair et compréhensible, et ce même dans plusieurs années. De plus, la documentation est une tâche qui forcera à commenter le code, et à écrire un code plus lisible et concis.

Attention, cependant cette opération demande du temps. Au final, c'est à chacun d'estimer si son projet a suffisamment de raison d'être documenté pour consacrer du temps à cette tâche.

Le wiki présente une manière d'utiliser une suite d'utilitaires simples pour produire une documentation.

Le Langage: Markdown

Le langage Markdown est un langage de balisage qui permet de créer facilement et rapidement de la documentation. Il est souvent utilisé pour de la documentation sur des sites tel que Github. Il peut également se convertir facilement en HTML, ce qui peut s'avérer utile.

Les créateurs de Markdown trouvaient que ce n'était pas vraiment pratique de devoir mettre en forme manuellement des textes en HTML, ils ont donc créé un petit langage très simple appelé le Markdown. L'idée est de pouvoir mettre en forme du texte sans avoir besoin de recourir à la souris... et sans avoir besoin de taper à la main des balises HTML toutes les 5 secondes.

Afin de ne pas devoir écrire et réécrire ma documentation avec différents outils pour ensuite les publier sous différentes formes j'utilise ce langage par balisage avec un simple éditeur de texte (GEDIT) pour produire des topics (court document) comportant un titre des sections et quelques paragraphes.

Ensuite les topics sont envoyés à divers programmes qui se chargeront de la mise en forme.

Les blogs et wikis.

dokuwiki

DokuWiki est un wiki Open Source simple à utiliser et très polyvalent qui n'exige aucune base de données. Il suffit de publier ses topics (en .txt), ou les dossiers contenant ces topics, dans le répertoire

data pour que ceux-ci soient immédiatement accessibles.

Le plugin **Markdowku** permet la prise en charge de Markdown.

gollum

Gollum est un wiki dont la particularité est d'être propulsé par git. Il a été codé par les développeurs de Github, le choix de git n'est donc pas étonnant. Il est ainsi possible d'éditer un wiki Gollum en passant par l'interface web ou directement depuis son éditeur de texte préféré, puis en commitant dans le dépôt git utilisé par Gollum.

Gollum est une application web sous licence MIT, développée en Ruby avec le framework Sinatra, et permettant:

- La prise en charge de plusieurs langages de markup (ASCIIDoc, Creole, **Markdown**, Org Mode...);
- La coloration syntaxique de bloc de code grâce à Pygments ;
- L'affichage de formules mathématiques au format TeX avec MathJax ;
- La possibilité d'attacher des fichiers et d'inclure des images.

Les éditeurs

Gedit

Le plugin **gedit-markdown** ajoute le support pour Markdown (ou Markdown Extra) dans gedit, l'éditeur de texte Gnome par défaut.

Plus précisément, il ajoute:

- Mise en évidence et extraits de syntaxe de Markdown;
- plugin Markdown Preview pour gedit, affiché dans le panneau latéral ou dans le panneau inférieur et prévisualisant en HTML le document ou la sélection en cours (ce plugin peut également être utilisé comme navigateur Web; voir la section Utilisation);
- un outil externe exportant au format HTML le document ou la sélection en cours;
- un jeu de couleurs, facultatif, mettant en évidence les fichiers de Markdown d'une manière plus similaire au rendu HTML.

Prérequis

gedit-markdown v2 prend en charge gedit 3.8 et 3.10. Il est livré avec un programme d'installation pour GNU / Linux.

Le plugin Markdown Preview dépend du paquet python3-markdown .

Pour les utilisateurs d'Ubuntu (et peut-être d'autres distributions) 11.10 ou ultérieure, le paquet

gir1.2-webkit-3.0 doit être installé pour utiliser le plugin Markdown Preview .

Installation (ou mise à jour)

1. Télécharger l'archive de gedit-markdown v2.
2. Extraire l'archive.
3. Ouvrir un terminal dans le dossier extrait.
4. Exécuter le programme d'installation dans le terminal:

```
./gedit-markdown.sh install
```

Le support de Markdown sera ajouté pour l'utilisateur actuel (donc pas besoin de privilèges root). Le dossier créé par l'extraction peut être supprimé après l'installation.

Désinstallation

1. Ouvrir un terminal dans le dossier extrait.
2. Exécuter le programme de désinstallation dans le terminal:

```
./gedit-markdown.sh uninstall
```

Vim

Le plugin Vimwiki ajoute un wiki personnel pour Vim - des fichiers texte simple liés entre eux et écrits dans un langage de balisage

Avec Vimwiki, on peut :

- organiser des notes et des idées
- gérer les listes de tâches
- écrire de la documentation
- tenir un journal
- tout exporter en HTML

Prérequis

S'assurer d'avoir ces paramètres dans le fichier vimrc:

```
set nocompatible
filetype plugin on
syntax on
```

Sans eux, Vimwiki ne fonctionnera pas correctement.

Installation à l'aide de packages Vim (depuis Vim 7.4.1528)

```
git clone https://github.com/vimwiki/vimwiki.git
~/.vim/pack/plugins/start/vimwiki
```

Les générateurs de documentation statiques

Rstudio

RStudio est un environnement de développement gratuit, libre et multiplateforme pour R, un langage de programmation utilisé pour le traitement de données et l'analyse statistique.

L'extension **rmarkdown** permet de générer des documents de manière dynamique en mélangeant texte mis en forme et résultats produits par du code R. Les documents générés peuvent être au format HTML, PDF, Word, et bien d'autres¹. C'est donc un outil très pratique pour l'exportation, la communication et la diffusion de résultats d'analyse.

Il permet également de produire une documentation dynamique à partir d'un document unique.

sphinx

Sphinx est un générateur de documentation libre. Il s'appuie sur des fichiers au format reStructuredText, qu'il convertit, entre autres, en HTML, PDF, EPUB, ou man.

Il supporte, entre autres, l'autogénération à partir du code source ou l'intégration de formules mathématiques.

Il supporte aussi l'auto-exécution de tests intégrés dans la documentation.

L'utilisation de l'extension **recommonmark** permet de générer des docs Docutils & Sphinx à partir de topics en Markdown.

La documentation ainsi générée peut être automatiquement déployée sur un serveur de pages statiques (ex. JEKYLl)

Pour produire une documentation avec sphinx

- 1. Créer le dossier source de la documention

```
$ mkdir madoc
```

- 2. Initier la documention

```
$ cd madoc
```

\$ sphinx-quickstart

- 3. Placer les topics dans le dossier
- 4. Générer la documentation en la publiant dans le dossier de **jekylls** contenant les pages statiques

```
sphinx-build -b html /data/git/static/madoc/  
/data/git/site/_pages/docs/madoc
```

shocco

Shocco est un générateur de documentation rapide et peu élaboré, conçu pour et dans le shell POSIX.

shocco lit les scripts shell et produit une documentation source annotée au format HTML. Les commentaires sont formatés avec Markdown et présentés à côté du code mis en évidence par la syntaxe afin de donner un effet d'annotation.

La documentation ainsi générée peut être automatiquement déployée sur un serveur de pages statiques (ex. Jekyll)

From:

<http://www.ouarte.garden/> - **dwndoc**

Permanent link:

<http://www.ouarte.garden/doku.php?id=activite:documentaion>

Last update: **2025/02/19 10:59**

