
Tutoriel Pandoc : MarkDown vers PDF

Ce document explique comment installer et utiliser l'outil Pandoc, dans le but de créer un PDF à partir d'un fichier texte Markdown.

Il a été écrit par les élèves de la formation Ingénierie des low-tech à l'Ecole Centrale de Nantes en Octobre 2023

Version	Mise à jour	License
v1.0	13/10/2023	Creative Commons Zero

Table des matières

Comment créer un PDF à partir de Markdown via Pandoc	3
Installation	3
Création d'un document	3
Edition d'un document	3
Page de titre	3
Corps du document	3
Images	4
Export PDF	4
Annexes	4
Arborescence du dossier Template	4
Contenu des fichiers textes	4
createPDF.bat	4
createTitlepagePDF.bat	4
titlepage.tex	4
template.md	5

Comment créer un PDF à partir de Markdown via Pandoc

Ce tuto résume et adapte celui qui m'a fait découvrir Pandoc : tuto de Guilwen Meunier.

Il contient une section *Annexes* dans laquelle se trouvent toutes les informations permettant de reconstituer facilement le dossier modèle.

Installation

1. Pandoc (tuto) et texlive (tuto).
Ouvrir un terminal et taper `pandoc --version` pour vérifier que Pandoc est installé, et `pdflatex` pour l'outil PDF que Pandoc utilise.
Guilwen recommande texlive-full (environ 5GB), disant qu'on peut tester et utiliser texlive-small ou texlive-basic. Je ne sais pas où se fait ce choix mais je n'en ai pas eu besoin.
2. Installer le package pdfpages : dans un terminal taper `tlmgr install pdfpages`.

Création d'un document

1. Copier le dossier `Template`, le renommer à volonté
2. Renommer le fichier `template.md` avec le nom souhaité en gardant l'extension `.md`
3. Clic droit sur le fichier `createPDF`, puis `Modifier`
4. Dans l'éditeur qui s'ouvre, remplacer le nom de fichier `template` par le nouveau nom choisi à l'étape 2 (ce changement dans l'éditeur se fait pour un fichier `.md` et pour un `.pdf`)

Edition d'un document

Page de titre

Dans le dossier `titlepage`, vous pouvez éditer le fichier `titlepage.tex`. Celui-ci est écrit en LaTeX. Même sans connaître ce langage, vous trouverez facilement où changer le titre, le sous-titre, ou bien les informations encadrées.

Une fois ce changement fait, double-cliquer sur le fichier de commande `createTitlepagePDF` pour l'exécuter. Si tout se passe bien, un terminal s'ouvre, exécute du code sans message d'erreur (seulement des messages de log), et vous laisse fermer la fenêtre. La page de titre au format PDF est alors actualisée dans le fichier `titlepage.pdf` que vous pouvez ouvrir pour vérifier le rendu.

Corps du document

L'édition se fait dans le fichier initialement appelé `template.md` que vous avez renommé.

Le début du document, entre les deux balises `---` sert à paramétrer l'affichage. Ces paramètres sont déjà prêts, pas besoin de comprendre ce qu'il y a là ! Seuls deux éléments dans les premières lignes peuvent nécessiter des modifications :

- le nom de l'auteur du document, écrit à la suite de `author` :
- la date du document, écrite après `date` :

Ces deux éléments n'apparaîtront pas sur le document, mais seront enregistrés comme métadonnées (consultables dans les propriétés ou détails du fichier).

Vous pouvez aussi choisir d'afficher (`true`) ou non (`false`) la table des matières (`toc`) et la liste des figures (`lof`) en modifiant le booléen écrit aux lignes correspondantes.

Après la deuxième balise `---`, le document commence. On trouve d'abord la commande LaTeX `\newpage` permettant de sauter une page après la table des matières (ou la liste des figures).

Et enfin, vous pouvez écrire librement en Markdown !

Images

Un point important à noter est le stockage des images : il est conseillé de les ranger dans le dossier `img`, comme le sont déjà les logos figurant en en-tête. Pour en afficher, il vous suffit alors décrire en Markdown (en remplaçant les textes et noms) :

```
![texte alt de l'image](img/nom_image.png "texte affiché au survol")
```

Export PDF

Double-cliquer sur le fichier de commande `createPDF` pour l'exécuter. Si tout se passe bien, un terminal s'ouvre, exécute du code sans aucun message, et vous laisse fermer la fenêtre quand il a fini. Le document complet au format PDF est alors actualisé dans le fichier `.pdf` que vous pouvez ouvrir pour vérifier le rendu, incluant la page de titre et le corps du document.

Annexes

Arborescence du dossier Template

- Template
 - `img`
 - * `logo-ECN.png`
 - * `logo-WELOW.png`
 - `titlepage`
 - * `createTitlepagePDF.bat`
 - * `titlepage.tex`
 - `createPDF.bat`
 - `template.md`

Contenu des fichiers textes

createPDF.bat

```
pandoc template.md -o template.pdf
pause
```

createTitlepagePDF.bat

```
pdflatex titlepage.tex
pause
```

titlepage.tex

```
\documentclass{article}

\usepackage[margin=30mm]{geometry}
\usepackage{varwidth}
\usepackage{array}
\setlength{\tabcolsep}{18pt}
\renewcommand{\arraystretch}{1.5}
\usepackage[export]{adjustbox}
```

```

\usepackage{pdfpages}
\usepackage{fancyhdr}
\pagestyle{fancy}

\begin{document}
  \begin{titlepage}
    \begin{center}
      \vspace*{1cm}
      \begin{figure}
        \begin{minipage}{0.48\textwidth}
          \includegraphics[height=1.4cm, left]{../img/logo-ECN.png}
          \vspace{2cm}
        \end{minipage}
        \hfill%
        \begin{minipage}{0.48\textwidth}
          \includegraphics[height=1.4cm, right]{../img/logo-WELow.png}
          \vspace{2cm}
        \end{minipage}
      \end{figure}
      \rule{\linewidth}{1pt} \\[0.4 cm]
      {
        \huge\bfseries
        \textbf{\Huge Titre du document}\\
        \par
      } \
      \rule{\linewidth}{1pt} \\[2.0 cm]
      \huge Sous-titre
      \vfill
      \large
      \begin{tabular}{|m{14cm}|}
        \hline
        \bigskip
        Ceci est un texte super important ou pas du tout, pouvant informer de la nature du do
        \bigskip\\
        \hline
      \end{tabular}
      \normalsize
      \vfill
    \end{center}
  \end{titlepage}
\end{document}

```

template.md

Metadata

title: #(leave title blank to prevent a page before your titlepage)

subtitle:

author: Option Ingénierie des low-tech Ecole Centrale de Nantes 2023-2024

date: 13/10/2023

lang: fr

```
# Parameters
geometry: margin=30mm
lof: false
toc: true
toc-depth: 3
toc-own-page: true
tof-own-page: true
indent: true
pagestyle: headings
colorlinks: true

# Title Page
include-before: |
    \includepdf[pages=-]{titlepage/titlepage.pdf}

# End Page
# include-after: |
#     \includepdf[pages=-]{img/endpage.pdf}

# Load packages & Define header/footer \fancyfoot[LE,RO]{\thepage}

header-includes: |
    \usepackage{varwidth}
    \usepackage{lastpage}
    \usepackage{tcolorbox}
    \definecolor{yback}{RGB}{242,242,242}
    \definecolor{yframe}{RGB}{172,147,147}
    \newtcolorbox{myquote}{colback=yback, colframe=yframe}
    \renewenvironment{quote}{\begin{myquote}}{\end{myquote}}
    \usepackage{pdfpages}
    \usepackage{sectsty}
    \usepackage{fancyhdr}
    \pagestyle{fancy}
    \setlength{\headheight}{11mm}
    \fancyhead[R]{\includegraphics[height=1.1cm]{img/logo-WELOW.png}}
    \fancyhead[L]{\includegraphics[height=1.1cm]{img/logo-ECN.png}}
    \fancyfoot[C]{Page\ \thepage\ sur\ \pageref{LastPage}}
---

\newpage

# Premier titre waouh
```