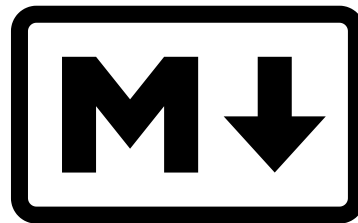




# Introduction à la prise de note avec markdown

Igor Milhit<sup>1</sup>



## Table des matières

1. <a href="#">Définition, avantages et inconvénients</a>	2
2. <a href="#">La syntaxe markdown</a>	4
3. <a href="#">Éditeurs</a>	6
4. <a href="#">Aperçu des possibilités avancées</a>	9
5. <a href="#">Bibliographie</a>	12



## Définition, avantages et inconvénients<sup>2</sup>

Le plus important est de trouver des méthodes et des outils adaptés à *ses besoins*. Ci-dessous sont listés quelques arguments en faveur de l'usage de fichiers au format texte et de la syntaxe markdown, mais aucune méthode ou outil n'est *la* solution miracle pertinente dans *tous* les contextes.

### Définition

L'information numérique est utile si elle est *structurée*. C'est ce qui permet, par exemple, que la table des matières d'un document numérique soit constituée de liens vers les sections. Cette structure peut s'obtenir par l'usage des styles et des niveaux de titre dans un traitement de texte (Microsoft Word, Libreoffice Writer, etc.), ou grâce à un langage de balisage comme HTML ou XML. Markdown est une autre méthode, très efficace.<sup>3</sup>

Markdown est *un langage de balisage léger*. Il s'agit d'une syntaxe qui permet, *uniquement avec des éléments textuels*, d'indiquer la structure (sections, titres) et la fonction d'éléments textuels (emphasis, liste, liens). Cette structure et ces fonctions sont lisibles, compréhensibles par des êtres humains<sup>4</sup> ou par des programmes informatiques. Aussi, les documents rédigés à l'aide de markdown peuvent soit être lus tels quels, soit être traités par un programme pour afficher un « rendu » sous forme de page web, de document PDF ou DOCX.

La syntaxe markdown est relativement simple à apprendre et mémoriser. Aussi, la rédaction au format markdown repose sur très peu de fonctionnalités de l'éditeur de texte. Le bloc note suffit, même si un peu plus de confort est souhaitable.

Cette syntaxe est utilisée par de nombreux logiciels ou services en lignes pour la rédaction. C'est le cas des notes dans [Zotero](#), de solution de prise de note comme [Joplin](#), [Notion](#) ou [Obsidian](#), ou encore d'éditeurs en ligne adaptés au travail collaboratif ([HedgeDoc](#), [HackMD](#), etc.).

### Avantages

Les fichiers utilisés ne sont composés que de caractères textuels. Aussi, ils sont :

- Simples et légers :
  - Ne dépendent pas d'un logiciel spécifique, ni d'un système d'exploitation.

- Pourront être ouverts et édités dans n'importe quel contexte, et probablement dans un avenir éloigné (pérennité).
- Rapides à ouvrir. Avec n'importe quel éditeur de texte (bloc note Windows, Notepad++, TextEdit, Gedit, VIM, EMACS, etc.)
- Faciles à sauvegarder, copier, synchroniser, partager.

Toujours grâce à leur nature textuelle, il est facile d'avoir des outils pour rechercher (et remplacer) rapidement du texte au sein non seulement du fichier ouvert, mais d'un ensemble de fichiers dans un répertoire.

Il est même possible d'écrire des programmes pour modifier ou utiliser automatiquement ces fichiers. Et il en existe justement beaucoup !

Grâce à ces propriétés, il est possible :

- De prendre des notes au clavier rapidement, de manière structurée, et avec une recherche rapide.
- De rédiger des documents simples (une note) ou complexes (une thèse).
- De regrouper un ensemble de notes dans un « bloc note » et de naviguer dans cet ensemble, de rechercher de l'information .
- De manière similaire, de regrouper un ensemble de fichiers pour constituer un document complexe, comme un livre ou un article scientifique.
- Enfin de convertir à partir des mêmes fichiers le document dans plusieurs formats pour une diffusion sur le web, la lecture dans un PDF ou un ePub (compatible avec une liseuse électronique).

Enfin, les fichiers textuels sont particulièrement adaptés au suivi de l'historique des modifications (*versioning*), par exemple avec le logiciel [Git](#).

## Inconvénients

Pour un usage simple, l'apprentissage n'est pas très difficile. La syntaxe de base s'acquiert rapidement. Pour parvenir à rédiger des documents relativement complexes (notes, citations et bibliographies, etc.), cet apprentissage est un peu plus long. Afin de maîtriser la conversion dans différents formats, tout en personnalisant la mise en forme des documents obtenus, il devient vite nécessaire d'apprendre à utiliser d'autres outils, à modifier des modèles dans les formats désirés (HTML, DOCX, peut-être LaTeX, etc.).

Certains éditeurs intègrent la fonction de conversion, mais elle repose sur l'installation de logiciels annexes, principalement `pandoc`, mais aussi une distribution LaTeX pour obtenir une conversion en PDF. Ces outils ne sont pas si difficiles à installer. Surtout, ils sont libres, gratuits et disponibles pour Windows, Mac OS ou Linux.

Pour utiliser `pandoc`, qui est l'outil de conversion le plus complet, il est nécessaire de le faire « en ligne de commande » plutôt que dans une interface graphique, avec la souris. C'est un élément en plus à apprendre, mais qui ouvre beaucoup de possibilités.

Le travail collaboratif à plusieurs est toujours possible, mais selon les cas avec une certaine friction. La situation la plus simple est d'utiliser un éditeur en ligne qui offre des fonctionnalités d'édition à plusieurs. Souvent cela est au prix de ne pouvoir disposer de toute la puissance de `pandoc`, du moins directement.

Toujours dans le cas où toutes les personnes qui collaborent sont capables de travailler avec les mêmes outils, elles peuvent toujours travailler ensemble en se mettant d'accord sur un « workflow », par exemple grâce à `git` mentionné plus haut. Mais c'est un outil de plus à apprendre. Et de nouvelles possibilités à découvrir.

Enfin, si les personnes avec qui on collabore ne veulent ou ne peuvent pas utiliser les mêmes méthodes, alors il faudra mettre en place un processus d'export et d'import un peu complexe, mais en partie automatisable ([Perret 2022b](#)).

## La syntaxe markdown

La syntaxe markdown est très bien documentée sur le web :

- La documentation « officielle » est disponible sur le site personnel d'un des concepteurs de la syntaxe : <https://daringfireball.net/projects/markdown/> ([Gruber 2004](#)).
- Le site *flus* propose une documentation en français et très claire : <https://flus.fr/carnet/markdown.html> ([Fressinaud 2022](#)).
- Il y a plusieurs variantes (ou extensions) de la syntaxe et un site essaie de proposer un standard. Il met à disposition un tutoriel interactif : <https://commonmark.org/help/> ([CommonMark](#)).
- Arthur Perret a traduit en français ce même tutoriel, disponible sur sa page expliquant ce qu'est markdown : <https://www.arthurperret.fr/cours/markdown.html> ([Perret 2022c](#)).

Pour les tableaux, il est fortement conseillé :

- D'utiliser un générateur, par exemple [https://www.tablesgenerator.com/markdown\\_tables](https://www.tablesgenerator.com/markdown_tables).
- Ou de disposer dans son éditeur d'un générateur de tableaux.

Le plus souvent les fichiers contenant du markdown ont pour extension `.md`, mais c'est une convention. Rien n'interdit d'utiliser d'autres extensions comme `.mkdn` ou

`.markdown`, pour autant que l'éditeur que vous utilisez les reconnaissent comme du markdown.

## Éditeurs

Il existe beaucoup d'éditeurs pour rédiger en markdown. On peut les regrouper en deux grandes catégories :

1. Les éditeurs [dédiés](#).
2. Les éditeurs [généralistes](#).

Dans les deux cas, des fonctionnalités de base qui sont presque incontournables :

- Une prévisualisation, le plus souvent sous la forme d'un *rendu* HTML. Cette fonctionnalité a l'avantage de permettre de se faire une idée du résultat sans devoir générer le fichier final.
- La possibilité de naviguer dans une arborescence de fichiers, en général dans une barre sur le côté de l'interface.
- Une autocomplétion, pour faciliter l'entrée de syntaxe avec des `[]` ou des `()`.

D'autres fonctionnalités ne sont pas indispensables, mais deviennent rapidement utiles :

- La correction orthographique pour les langues dans lesquelles vous rédigez.
- Des raccourcis pour la syntaxe (taper `img` et obtenir la syntaxe pour l'insertion d'une image).
- Un formatage automatique des tableaux.
- La possibilité de suivre les liens internes, au sein du même fichier ou entre les fichiers d'un même répertoire.

### Éditeurs dédiés

Ce sont des logiciels conçus spécifiquement pour éditer des fichiers avec la syntaxe markdown. Ils peuvent être sous la forme d'un logiciel à installer sur votre appareil (ordinateur, tablette, téléphone) ou d'un service web.

**Typora** est éditeur qui met l'accent sur son interface épurée. Il est disponible pour tous les systèmes d'exploitation, payant. <https://typora.io/>

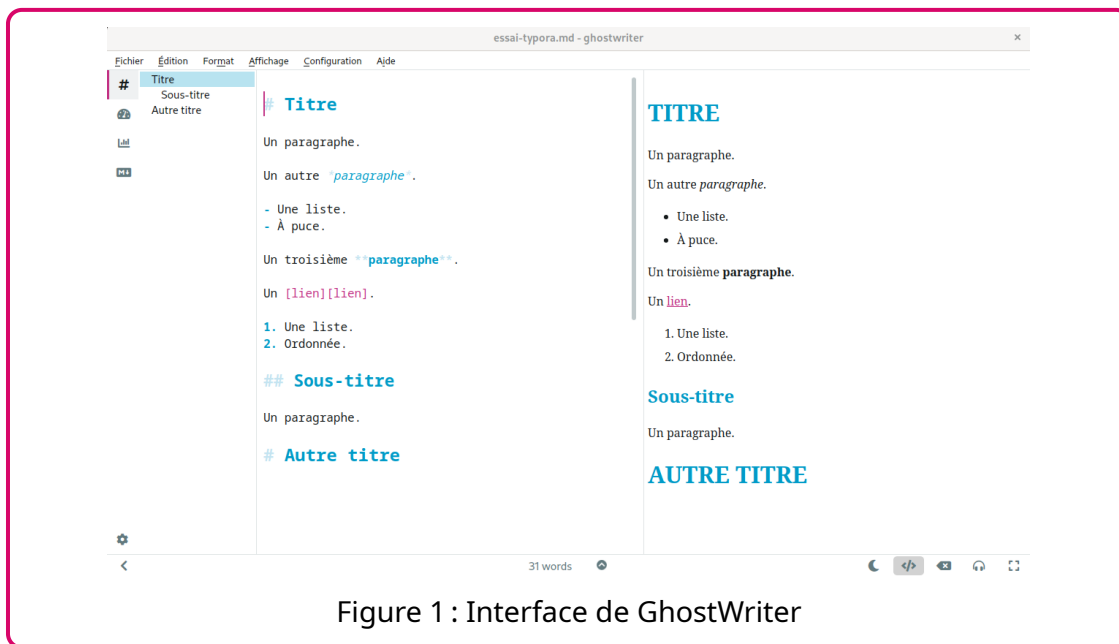


Figure 1 : Interface de GhostWriter

**GhostWriter** est libre et gratuit, pour Windows et Linux principalement.<sup>5</sup>

<https://ghostwriter.kde.org/fr/>

**Abricotine** est libre et gratuit, pour toutes les plateformes. <http://abricotine.brrd.fr/>

**ia Writer** est payant, pour MacOS et Windows, mais est très apprécié pour la rédaction. Il permet de lier des notes entre elles et offre ainsi une navigation dans un ensemble de fichiers. <https://ia.net/writer>

**Obsidian** est un logiciel propriétaire, pour toutes les plateformes. Il ne se limite pas à l'édition de fichiers markdown, mais permet de constituer une base de connaissance basée sur des notes, et cela de manière très complète. <https://obsidian.md/>

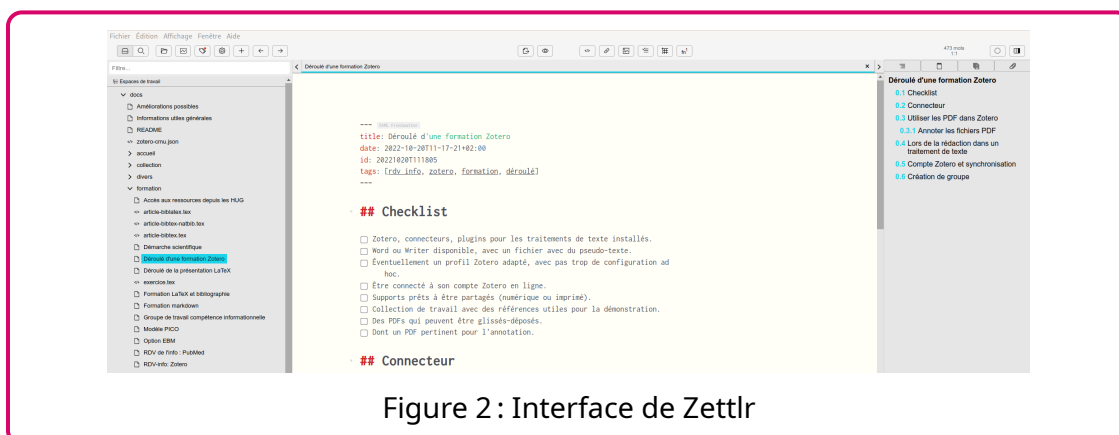


Figure 2 : Interface de Zettlr

**Zettlr** est libre et disponible pour tous les systèmes d'exploitation. Il est destiné à un usage académique, que ce soit pour construire une base de connaissance à partir de

notes ou pour rédiger un travail de mémoire ou un article scientifique. Il intègre Zotero pour la gestion des références.

En ligne, on peut mentionner à nouveau le service [HackMD](#) ou le logiciel [HedgeDoc](#) qui permet non seulement d'éditer des fichiers markdown avec une prévisualisation, mais également de le faire à plusieurs, de disposer d'un suivi des modifications et de commenter.

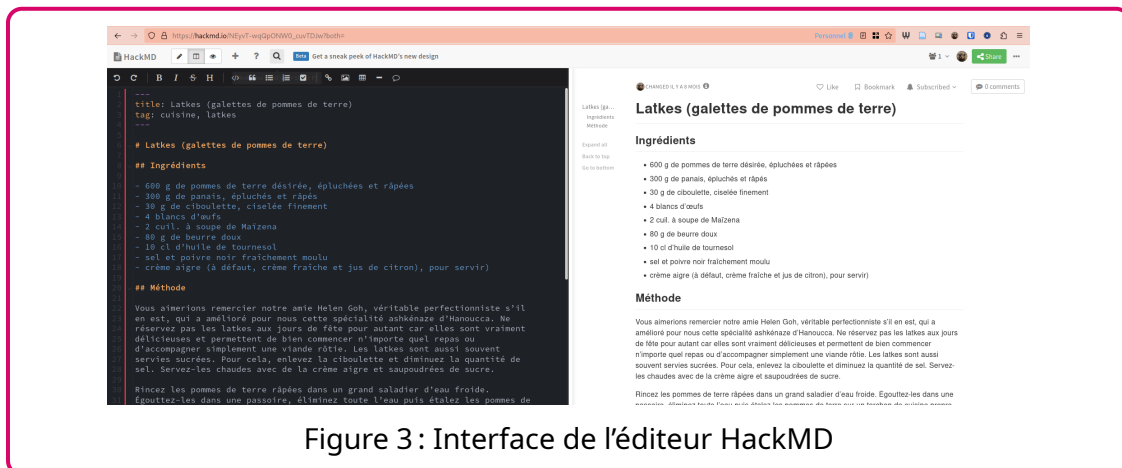


Figure 3 : Interface de l'éditeur HackMD

## Éditeurs généralistes

*Généralistes* car ces éditeurs ne sont pas limités à l'édition de fichier markdown. Le plus souvent, ce sont des *éditeurs de texte* utilisés par des développeurs pour écrire du code informatique. Le plus souvent, ils disposent d'un écosystème d'extensions afin de s'adapter à des besoins spécifiques. De ce fait, ils sont très puissants.

Ainsi, il est possible d'adapter à l'édition du markdown les éditeurs de texte suivant, notamment :

- Notepad++, libre et gratuit, pour Windows.
- Visual Studio Code, gratuit, pour Windows, MacOS, Linux. Un des logiciels les plus utilisés actuellement.
- VIM ou neovim, libre et gratuit, pour tous les systèmes d'exploitation.
- EMACS, libre et gratuit.



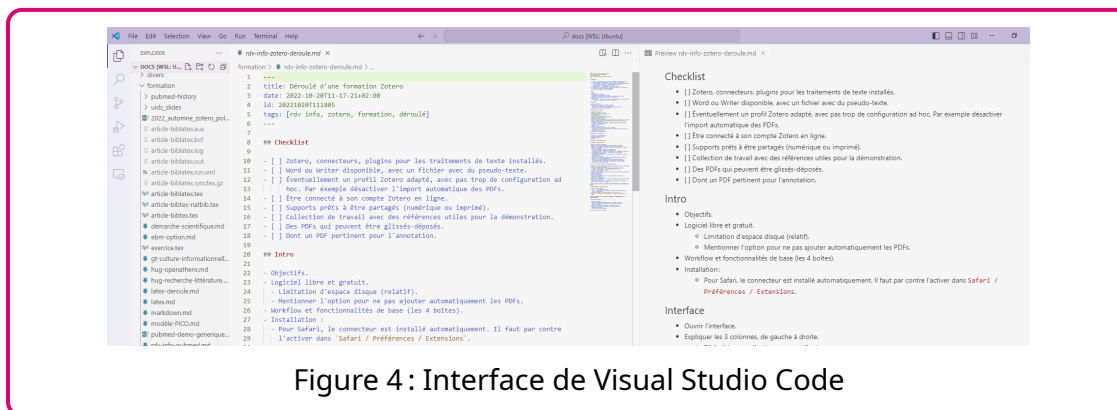


Figure 4 : Interface de Visual Studio Code

## Aperçu des possibilités avancées

### Ensemble de notes

Afin d'organiser ses notes, il est possible de réunir un ensemble de fichiers dans le même répertoire (y compris avec des sous répertoires) et de faire des liens entre les fichiers (et même au sein d'un seul fichier).

Certains éditeurs, ou certaines extensions d'éditeurs, proposent des facilités pour générer ces liens, au prix d'une syntaxe *ad hoc* (« wiki links »). Mais il est parfaitement possible de le faire avec la syntaxe « normale » du lien.



Figure 5 : Lien depuis le fichier README.md vers le fichier points-a-ameliorer.md

La figure 5 montre sur la gauche une arborescence de fichiers, avec des sous-répertoires, et dans le fichier affiché (README.md), un lien vers le fichier points-a-ameliorer.md. Dans cet éditeur, en plaçant le curseur sur le lien et en tapant la touche « entrée », on ouvre le fichier cible.

La figure 6 montre la recherche du terme `rdv-info` lancée sur la même arborescence de fichiers que la figure 5, au moyen du logiciel Visual Studio Code. Il y a 13 occurrences du terme dans 7 fichiers. Ce type de recherche est facilité par le fait que ce sont des fichiers au format texte et se retrouve dans la plupart des éditeurs généralistes de qualité.

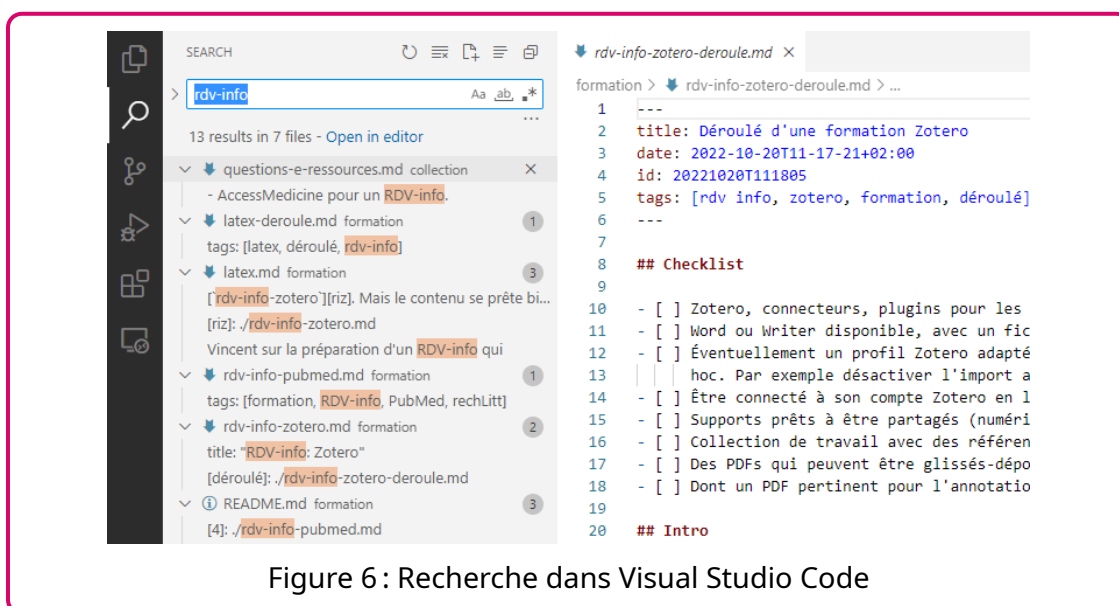


Figure 6 : Recherche dans Visual Studio Code

## Export vers d'autres formats

La conversion d'un document markdown en un autre format, plus simple à partager devient assez vite utile. Pour les conversions, `pandoc` est l'outil le plus souvent utilisé. Il est soit utilisé par votre éditeur pour faire les exports (par exemple avec Zettlr), soit par vous même en ligne de commande. C'est un logiciel libre et gratuit, développé très activement.

Il est capable de lire et d'écrire dans un grand nombre de formats différents, parfois à l'aide de logiciels spécifiques. Par exemple, pour produire un fichier PDF, il peut utiliser plusieurs méthodes, dont LaTeX, ce qui suppose d'en avoir une version installée.

La commande pour une conversion simple est de la forme suivante :

```
pandoc --output fichier.docx fichier.md
```

Le format obtenu correspond à celui qui est indiqué en extension du nom de fichier de sortie `--output`, ici du DOCX. Mais il est possible de l'explicitier :

```
pandoc --to docx --output fichier.docx fichier.md
```

Le support de cours de cette introduction à markdown a été rédigée en markdown et est exportée en PDF grâce à `pandoc` et à *Paged.js*.<sup>6</sup>

## Citations et Bibliographie

Pour insérer des citations et générer une bibliographie, il est nécessaire d'avoir au moins une collection de références bibliographiques, par exemple sous la forme d'un fichier `.bib`. Pour obtenir ce fichier, il est possible de passer par Zotero et d'exporter une collection dans le format BibTex (ou Better BibTex avec l'extension Zotero du même nom.)

Dans ce fichier (et dans Zotero), on trouve pour chaque référence une « clé de citation » qui permet de l'identifier de manière unique. C'est cette clé que l'on va insérer dans notre fichier markdown pour indiquer une citation. Elle est insérée avec la syntaxe suivante: `[@clé-de-citation]`.

On peut encore ajouter à la fin de notre fichier un titre de deuxième niveau avec le texte bibliographie, pour que `pandoc` puisse y insérer la bibliographie.

Pour être en mesure de traiter ces informations, il faut utiliser l'option de `--citeproc` intégrée dans `pandoc` :

```
pandoc --citeproc \  
  --to=pdf --pdf-engine=xelatex \  
  --bibliography=references.bib \  
  --output fichier.pdf fichier.md
```

Dans l'exemple ci-dessus, nous voyons l'option `--citeproc` et l'option `--bibliography=` qui indique où se trouve le fichier des références bibliographiques. Par défaut le style est Chicago, de type auteur-date.

Pour aller plus loin, le mieux est de consulter la [documentation officielle](#) de `pandoc`.

## Bibliographie

COMMONMARK. CommonMark. *CommonMark* [en ligne]. Disponible à l'adresse : <https://commonmark.org/> [consulté le 25 avril 2023].

DELETRAZ, Gaëlle et RABAUD, Julien, 2022. *En mode texte : Markdown, Stylo, Pandoc, Notebook...* [en ligne]. Atelier. Pau. 2022. Disponible à l'adresse : <https://markdown-somate2022.netlify.app> [consulté le 17 avril 2023].

EYSSETTE, Cédric, 2023. *Utiliser le markdown pour tout faire* [en ligne]. 2023. Disponible à l'adresse : <https://eyssette.forge.aeif.fr/marp-slides/slides/2022-2023/utiliser-le-markdown-pour-tout-faire> [consulté le 17 avril 2023].

FRESSINAUD, Marien, 2022. Guide Markdown. *flus* [en ligne]. 11 juillet 2022. Disponible à l'adresse : <https://flus.fr/carnet/markdown.html> [consulté le 17 avril 2023].

GRUBER, John, 2004. Markdown. *Daring Fireball* [en ligne]. 17 décembre 2004. Disponible à l'adresse : <https://daringfireball.net/projects/markdown/> [consulté le 17 avril 2023].

MARTINOLLI, Pascal, 2024. *Manuel pratique de recherche documentaire* [en ligne]. Disponible à l'adresse : <https://pmartinolli.github.io/QMpRD/> [consulté le 12 novembre 2024].

MASUTTI, Christophe, 2019. Rédiger et manipuler des documents avec Markdown. *Stadium : blog* [en ligne]. 8 mai 2019. Disponible à l'adresse : <https://golb.stadium.link/post/20190528rediger-et-manipuler-des-documents-markdown/> [consulté le 19 mars 2024].

OVADIA, Steven, 2014. Markdown for Librarians and Academics. *Behavioral & Social Sciences Librarian*. Vol. 33, no 2, pp. 120-124. DOI [10.1080/01639269.2014.904696](https://doi.org/10.1080/01639269.2014.904696).

PAGED.JS. About Paged.js? *Paged.js* [en ligne]. Disponible à l'adresse : <https://pagedjs.org/about/> [consulté le 8 mars 2024].

PERRET, Arthur, 2022a. Format texte. *Arthur Perret* [en ligne]. Disponible à l'adresse : <https://www.arthurperret.fr/cours/format-texte.html> [consulté le 18 avril 2023].

PERRET, Arthur, 2022c. Markdown. *Arthur Perret* [en ligne]. Disponible à l'adresse : <https://www.arthurperret.fr/cours/markdown.html> [consulté le 17 avril 2023].

PERRET, Arthur, 2022b. Markdown et traitement de texte : une méthode pour collaborer. *arthurperret.fr* [en ligne]. 25 octobre 2022. Disponible à l'adresse : <https://www.arthurperret.fr/blog/2022-10-25-markdown-traitement-de-texte-methode-pour-collaborer.html> [consulté le 23 avril 2024].

POLLARD, Tom, 2023. *Template for writing a PhD thesis in Markdown* [logiciel] [en ligne]. 17 avril 2023. [consulté le 17 avril 2023]. Disponible à l'adresse : [https://github.com/tompollard/phd\\_thesis\\_markdown](https://github.com/tompollard/phd_thesis_markdown) [consulté le 17 avril 2023].

## Notes

---

1. Uni CMU ↩
2. Cette section repose beaucoup sur la page [Format texte](#) d'Arthur Perret ([Perret 2022a](#)). ↩
3. Markdown n'est pas le seul langage de balisage léger existant, bien que ce soit le plus répandu. D'autres sont encore plus robustes ou puissants, comme *RestructuredText* ou *AsciiDoc*. ↩
4. Au moins si on compare la syntaxe Markdown au HTML ou au XML. ↩
5. Il semble possible de l'installer sur MacOS, mais n'est pas recommandé. ↩
6. Les sources et la méthode de conversion est disponible sur <https://git.milhit.ch/igor/rdv-info-markdown>. ↩