Markdown

One Template to rule them all

A template for all your markdown needs. Creating beautiful documents and slides by only using markdown.

github.com/uss-zerodata

01.01.2024

Contents

1 Markdown Template

One Template to rule them all A template for all your markdown needs. Creating beautiful documents written in markdown and rendered using pandoc.

Document Example

Figure 1: Document Example

Document Example PDF

Slides Example (Dark)

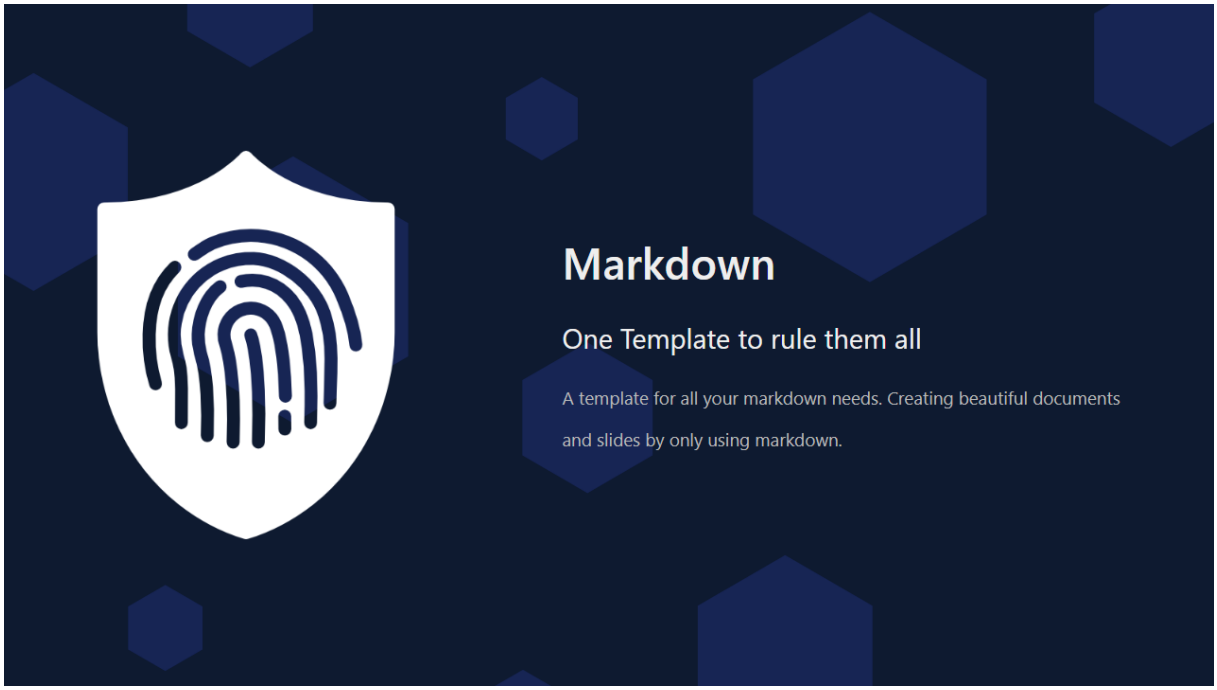

Figure 2: Dark Slides Example

Dark Slides Example PDF

Slides Example (Light)

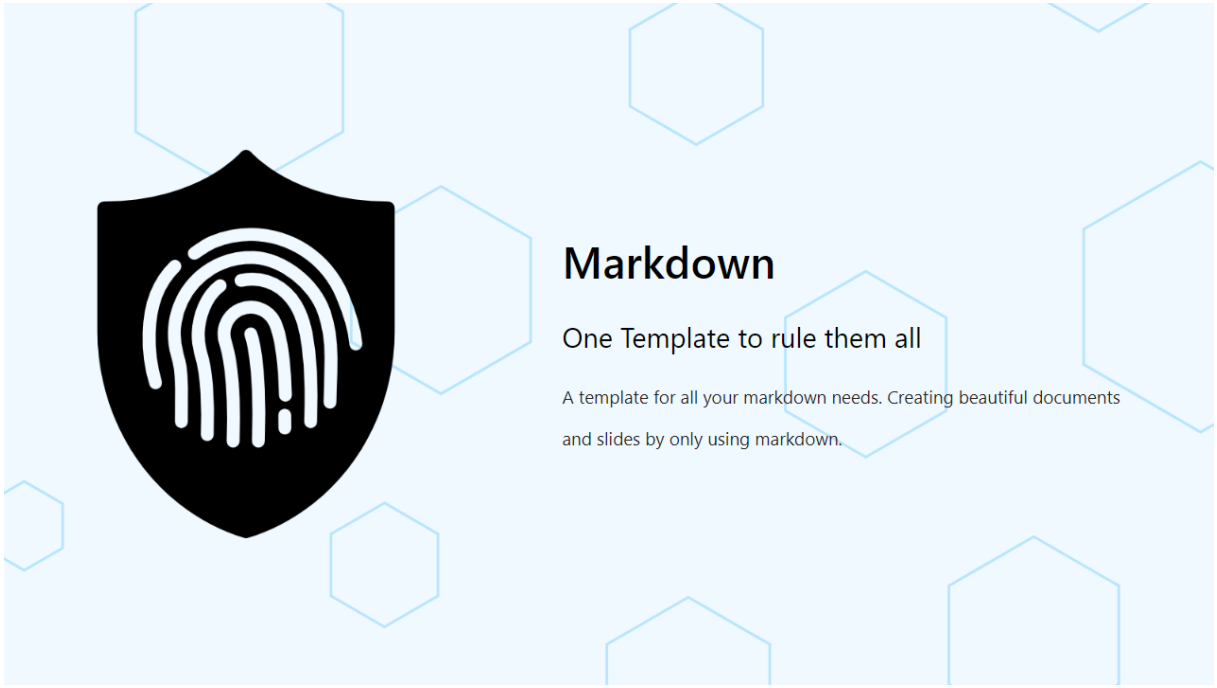

Figure 3: Light Slides Example

Light Slides Example PDF

2 Features

- Beautiful title page
- Table of contents
- Bibliography
- Full markdown support
- Easy to use
- Easy to extend

3 Usage

Write your document in the `main.md` file. Then compile it using the Makefile. You can use all features of markdown in this file. For a detailed description of the markdown syntax see the markdown guide or the markdown cheat-sheet.

In addition to the markdown syntax you can also use LaTeX commands in your document. For example you can use `\newpage` to enforce a page break or `\cite` to quote a source from a bibliography file. For a detailed description of the LaTeX syntax see the LaTeX Wikibook or the LaTeX cheat-sheet.

Presentations can be created in the `slides.md`. In contrast to the main document, it does not support LaTeX commands. But you can use all Markdown features in combination with CSS.

3.1 Compiling

After you finished writing your document you can compile it to a PDF file using `make`.

```
make            # Renders the main.md and slides.md files to various formats. (Default)
make document     # Renders the main.md file to a PDF document.
make slides     # Renders the slides.md file to a PDF, HTML and PPTX file.
make clear      # Deletes all temporary files
make purge      # Deletes all temporary and output files
make setup      # Installs the template and all dependencies
make help       # Shows a help message with all available commands
```

4 Setup

To automatically install all dependencies use the Makefile.

```
make setup
```

Reccomended Editor

For editing, we recommend using Visual Studio Code. It is a free and open source text editor, that can be customized with extensions to fit your needs. We recommend the following extensions.

- Code Spell Checker
- Excel to Markdown table
- Image preview
- Markdown All in One
- Markdown Shortcuts
- Markdown+Math
- Marp for VS Code
- Prettier - Code formatter
- Todo Tree

4.1 Windows

If you are using Windows you can use Chocolatey[1] to install the dependencies through the command line. If you don't have Chocolatey installed already, you can install it by following the instructions on the Chocolatey website[1].

```
# Chocolatey
choco install pandoc miktex strawberryperl make nodejs python3
npm install

# Unzip document backgrounds
unzip -o assets/ assets/a4.zip

# Generate slide backgrounds
cd assets/textures
python3 generate_textures.py
```

Manual Installation

It is also possible to install the dependencies manually.

- Pandoc[2]
- MiKTeX[3]
- Strawberry Perl[4]
- Make[5]
- Node.js
- Python 3

4.2 Linux

If you are using Linux you can use the package manager of your distribution to install the dependencies. The following examples are for Debian based distributions like Ubuntu.

```
# Debian/Ubuntu
sudo apt update
sudo apt-get install pandoc texlive-full perl make nodejs python3
npm install

# Unzip document backgrounds
unzip -o assets/ assets/a4.zip

# Generate slide backgrounds
cd assets/textures
python3 generate_textures.py
```

Manual Installation

It is also possible to install the dependencies manually.

- Pandoc[2]
- TeX Live[6]
- Perl[7]
- Make[8]

- Node.js
- Python 3

5 Customization

Pandoc Options

5.1 Pandoc Options

You can customize the template by changing the pandoc options in the Makefile. Some examples are given below.

```
pandoc -o output.pdf input.md --template assets/eisvogel.tex

# favorite template for general purpose
pandoc -o output.pdf .main.md --template assets/eisvogel.tex --listings

# optional highlight style to use instead of --listings
# to get a list of all available styles, use: pandoc --list-highlight-styles
pandoc -o output.pdf .main.md --template assets/eisvogel.tex  --highlight-style kate

# favorite template for bigger scientific documents
pandoc -o output.pdf .main.md --template assets/eisvogel.tex --listings --toc --toc-depth 2 --number-sections

# special options for setting a book
pandoc -o output.pdf .main.md --template assets/eisvogel.tex --listings --toc --toc-depth 2 --number-sections -V book -V classoption=oneside --top-level-division=chapter
```

- `-o output.pdf`: output file is output.pdf
- `input.md`: input file is input.md
- `--template assets/eisvogel.tex`: use eisvogel.tex as template
- `--listings`: use listings package for code highlighting
- `--top-level-division=chapter`: use chapters instead of sections
- `--toc`: generate table of contents
- `--toc-depth 2`: set depth of table of contents to 2

Template Variables

5.2 Template Variables

You can customize the template by changing the front matter of the `main.md` file. The following options are available.

- `titlepage` (defaults to **`false`**)

  turns on the title page when **`true`**

- `titlepage-color`

  the background color of the title page. The color value must be given as an HTML hex color like `D8DE2C` without the leading number sign (#). When specifying the color in YAML, it is advisable to enclose it in quotes like so `titlepage-color: "D8DE2C"` to avoid the truncation of the color (e.g. `000000` becoming `0`).

- `titlepage-text-color` (defaults to `5F5F5F`)

  the text color of the title page

- `titlepage-rule-color` (defaults to `435488`)

  the color of the rule on the top of the title page

- `titlepage-rule-height` (defaults to `4`)

  the height of the rule on the top of the title page (in points)

- `titlepage-logo`

  path to an image that will be displayed on the title page. The path is always relative to where pandoc is executed. The option `--resource-path` has no effect.

- `titlepage-background`

  the path to a background image for the title page. The background image is scaled to cover the entire page. In the examples folder under `titlepage-background` are a few example background images.

- `page-background`

  the path to a background image for any page. The background image is scaled to cover the entire page. In the examples folder under `page-background` are a few example background images.

- `page-background-opacity` (defaults to `0.2`)

  the background image opacity

- `caption-justification` (defaults to `raggedright`)

  justification setting for captions (uses the `justification` parameter of the caption package)

- `toc-own-page` (defaults to **`false`**)

  begin new page after table of contents, when **`true`**
- `listings-disable-line-numbers` (defaults to **`false`**)

  disables line numbers for all listings
- `listings-no-page-break` (defaults to **`false`**)

  avoid page break inside listings
- `disable-header-and-footer` (default to **`false`**)

  disables the header and footer completely on all pages
- `header-left` (defaults to the title)

  the text on the left side of the header
- `header-center`

  the text in the center of the header
- `header-right` (defaults to the date)

  the text on the right side of the header
- `footer-left` (defaults to the author)

  the text on the left side of the footer
- `footer-center`

  the text in the center of the footer
- `footer-right` (defaults to the page number)

  the text on the right side of the footer
- `footnotes-pretty` (defaults to **`false`**)

  prettifies formatting of footnotes (requires package `footmisc`)
- `footnotes-disable-backlinks` (defaults to **`false`**)

  disables making the reference from the footnote at the bottom of the page into a link back to the occurence of the footnote in the main text (enabling requires package `footnotebackref`).
- `book` (defaults to **`false`**)

  typeset as book
- `logo-width` (defaults to `35mm`)

  the width of the logo. One needs to specify the width with a (TeX) unit e.g. `100pt` or `35mm`. The following units can be used:

  - `pt`: Point
  - `pc`: pica (12 `pt`)
  - `in`: inch (72.27 `pt`)
  - `bp`: Big point (72 `bp` = 1 `in`)
  - `cm`: Centimeter
  - `mm`: Millimeter
  - `dd`: Didot point
  - `cc`: cicero (12 `dd`)
  - `sp`: Scaled point (65,536 `sp` = 1 `pt`)
  - `ex`: Nomimal x-height
  - `em`: Nominal m-width
  - `px`: Pixel (only for pdfTeX and LuaTeX) The dimension given to the `\pdfpxdimen` primitive; default value is 1 `bp`, corresponding to a pixel density of 72 dpi.

  A visual overview of the length units is available at https://github.com/tweh/tex-units.

- `first-chapter` (defaults to 1)

  if typesetting a book with chapter numbers, specifies the number that will be assigned to the first chapter

- **`float`**`-placement-figure` (defaults to `H`)

  Reset the default placement specifier for figure environments to the supplied value e.g. `htbp`. The available specifiers are listed below. The first four placement specifiers can be combined.

  1. `h`: Place the float *here*, i.e., approximately at the same point it occurs in the source text.
  2. `t`: Place the float at the *top* of the page.
  3. `b`: Place the float at the *bottom* of the page.
  4. `p`: Place the float on the next *page* that will contain only floats like figures and tables.
  5. `H`: Place the float *HERE* (exactly where it occurs in the source text). The `H` specifier is provided by the float package and may not be used in conjunction with any other placement specifiers.

- `table-use-row-colors` (defaults to **`false`**)

  enables row colors for tables. The default value is **`false`** because the coloring extends beyond the edge of the table and there is currently no way to change that.

- `code-block-font-size` (defaults to `\small`)

  LaTeX command to change the font size for code blocks. The available values are `\tiny`, `\scriptsize`, `\footnotesize`, `\small`, `\normalsize`, `\large`, `\Large`, `\LARGE`, `\huge` and `\Huge`. This option will change the font size for default code blocks using the verbatim environment and for code blocks generated with listings.

6 Credits

- Pandoc
- LaTeX
- Eisvogel Template

References

[1] choco, “Chocolatey - The package manager for Windows.” [Online]. Available: https://chocolatey.org/

[2] Pandoc, “Pandoc - Installing pandoc.” [Online]. Available: https://pandoc.org/installing.html

[3] MiKTeX, “Getting MiKTeX.” [Online]. Available: https://miktex.org/download

[4] Perl, “Strawberry Perl for Windows.” [Online]. Available: https://strawberryperl.com/

[5] GNUwin, “Make for Windows.” [Online]. Available: https://gnuwin32.sourceforge.net/packages/make.htm

[6] TeXLive, “TeX Live - TeX Users Group.” [Online]. Available: https://www.tug.org/texlive/

[7] Perl, “Perl Download - www.perl.org.” [Online]. Available: https://www.perl.org/get.html

[8] GNU, “Make - GNU Project - Free Software Foundation.” [Online]. Available: https://www.gnu.org/software/make/