MM-DOCS

This project is used to produce all forms of documentation for the service using Markdown and other repository friendly markup languages.

The documentation is created using the mm-docs docker bundle:

https://github.com/majkinetor/mm-docs

Features

- Write documentation in extended Markdown, HTML, PlantUML and other markup languages
- · Beautiful responsive design using material theme
- Navigational bar, TOC, search, internationalization, and various rich presentation functionalities
- Rich diagrams and interface mockups using PlantUML
- Macros can be written in python
- PDF export for each page or full site
- Live reload during editing
- GitLab/GitHub repository connection for editing within a browser
- Automated AppVeyor build and deploy to GitHub pages using Ubuntu image
- · Cross platform for hosting and development

Prerequisites

- Docker
- PowerShell (optional, but recommended)

© 2019 majkinetor mm-docs demo

Quick start

To edit documentation:

- Clone new project template
- Start development server run in PowerShell Invoke-Build Run . The command is syntax sugar for running docker containers.
- Add new documents somewhere in the source\docs hierarchy. To be visible in navigation,
 add to the path to the file in source\mkdocs.yml section nav, otherwise, the page is
 available via direct link.

To build static site run Invoke-Build Build.

This is a footer page, edit in inc/footer.md

Demonstration page

Date:

mm-docs demonstration page.

For more details about included tools see:

- Mkdocs
- Mkdocs Material Extensions
- PlantUML
- MathJax

Customization

- Color palettes
- Language
- Extending theme

mm-docs demo

Text

Basic effects

Code	Output
Bold text orBold text	Bold text
Italic text or _Italic text_	Italic text
~~Strike Through~~	Strike Through
==Colored text==	Colored text
`Inline code`	Inline code

Hotkeys



• Unicode: Мислим дакле постојим

Links, footnotes and comments

Links

- I'm an inline-style link
- I'm an inline-style link with title

mm-docs demo

© 2019 majkinetor

- I'm a reference-style link
- · I'm a relative reference to a repository file
- You can use numbers for reference-style link definitions
- You can leave a link empty and use the link text itself

Magic links and emails: turned to links as recognized

- http://www.google.com
- majkinetor@gmail.com
- www.google.com

Footnotes

- I am a text with a short footnote |1
- I am a text with a long footnote |2

Comments

The text contains 2 comments bellow this line which should not be visible.

Citations

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa. Inline:

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

code in citations

Another citation line

HTML

HTML is allowed without restrictions:

Some big text

Critic and comments

Here is some incorrect Markdown. I am adding this here. Here is some more text that I am adding. Paragraph was deleted and replaced with some spaces.

Spaces were removed and a paragraph was added.

And here is a comment on some text This works quite well. I just wanted to comment on it. .

Substitutions is are great!

General block handling:

- test remove
- test remove
- test add
- test add

Emoji

List of emoji codes

- :tada: 🎇
- :jack_o_lantern: 🎃
- :alien: 🕹

- :robot_face: 👜
- :smile: 😄
- :heart: 💚
- :thumbsup: 👍
- ♥! Ø # ¥ \$



Ordered list

- 1. List item 1
- 2. List item 2
 - a. List item 2.1
 - b. List item 2.2
 - i. List item 2.2.1
 - ii. List item 2.2.2
 - c. List item 2.3
- 3. List item 4

Unordered, mixed

- List item 1
- List item 2
 - List item 2.1
 - List item 2.2
 - a. List item 2.2.1
 - b. List item 2.2.2
 - List item 2.3
- List item 4

Tables

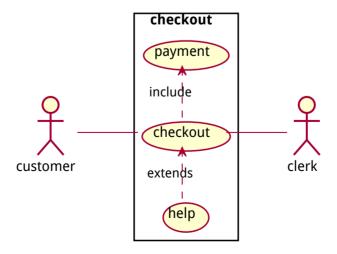
Col1	Col2
row1.1	row1.2
row2.1	row2.2-1 row2.2-2

Plantuml

Add plantuml in a block code header to set code section as PlantUml source. Use format to specify output type:

```
'``plantuml format="svg"
title PlantUml Demo
left to right direction
skinparam packageStyle rectangle
actor customer
actor clerk
rectangle checkout {
  customer -- (checkout)
  (checkout) .> (payment) : include
  (help) .> (checkout) : extends
  (checkout) -- clerk
}
'``
```

PlantUml Demo



mm-docs demo

MathJax

The block code:

```
$$
\frac{n!}{k!(n-k)!} = \binom{n}{k}
$$
...
```

 $\frac{n!}{k!(n-k)!} = \frac{n}{k!}$

```
Inline math expressions: p(x|y) = \frac{p(y|x)p(x)}{p(y)}
```

Inline math expressions: $p(x|y) = \frac{p(y|x)p(x)}{p(y)}$

Admonition

More info



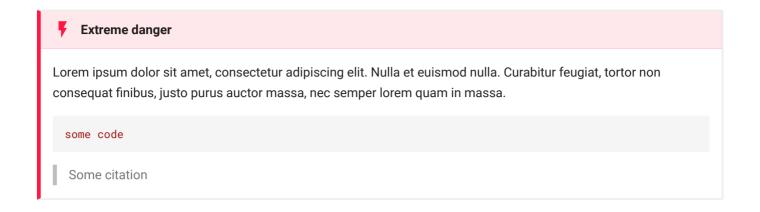
Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

Collapsed closed

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.

? Collapsed open

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod nulla. Curabitur feugiat, tortor non consequat finibus, justo purus auctor massa, nec semper lorem quam in massa.



Code Highlight

```
import tensorflow as tf
```

With tabs:

```
Bash PowerShell UML C C++ C#

#!/bin/bash
echo "Hello world!"
```

Abbreviations

The following are defined in single file, _inc\abbr.md and included on each page.

• IT, XML, JSON

Smart symbols

- (tm) **
- (c) ©
- · c/o %
- +/- ±
- --> →

- <-- ←
- <--> **↔**
- =/= ≠
- 1/4 1/4
- 1st 1st

Task list

- Lorem ipsum dolor sit amet, consectetur adipiscing elit
- Curabitur elit nibh, euismod et ullamcorper at, iaculis feugiat est
- ☐ Vestibulum convallis sit amet nisi a tincidunt
 - In hac habitasse platea dictumst
 - Praesent sed risus massa
- ☐ Nulla vel eros venenatis, imperdiet enim id, faucibus nisi

Custom span class

Adding custom HTML classes to span elements.

```
Syntax: !!<class names>|<text to be wrapped>!!.
```

The following example uses css from the custom_css file, classes color-yellow and backgrond-blue:

• I love !!color-yellow background-blue|cookies!! - I love cookies

Macros

More details.

Variable from config	extra.version = 0.1
Variable from this page	acme = Acme Company Ltd
Macro (defined in main.py)	bar(1) = 9.3 - barbaz(3) = 9
Python variable (defined in main.py)	baz = John Doe
Dictionary variable from config	company.name = Acme

Includes

Include other markdown files via 2 mechanisms:

- Via macros plugin
- Via markdown_include plugin

Macros plugin is much better as it recursively handles variables in partials.

Environment variables

Environment variables can be used inside of configuration file mkdocs.yaml .

```
site_name: !!python/object/apply:os.getenv ["CI_PROJECT_PATH"]
repo_url: !!python/object/apply:os.getenv ["CI_PROJECT_URL"]
repo_name: !!python/object/apply:os.getenv ["CI_PROJECT_PATH"]
```

- 1. https://en.wikipedia.org/wiki/Note_(typography)
- 2. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Nulla et euismod massa, nec semper lorem quam in massa.

```
code
```