Create **beautiful** and **localized** documentations and websites using **MkDocs + Github**
Bonjour !

**Open Source community service**

**Open Source author & contributor**
- Author of py3status, uhashring, mkdocs-static-i18n
- MkDocs contributor & maintainer
- PSF contributing member

**Gentoo Linux developer**
- Clustering related packages
- Gentoo Docker containers

**Tech speaker & writer**
- Tech conferences, webinars, blog posts
- https://ultrabug.fr
MkDocs is a fast, simple and downright gorgeous static site generator that's geared towards building project documentation.

Documentation source files are written in Markdown, and configured with a single YAML configuration file.

MkDocs core is written in Python, its source code is easy to understand and we are open to contributions!

https://mkdocs.org
https://www.markdownguide.org/
https://www.markdowntutorial.com/
**mkdocs-material** is a **Material Design theme for MkDocs** that allows you to create a branded static site from a set of Markdown files to host the documentation of your Open Source or commercial project.

It is **customizable, searchable, mobile-friendly and supports 40+ languages**.

https://squidfunk.github.io/mkdocs-material/

https://material.io/
A benevolent debate on i18n vs l10n took place on the 774 where we settled that internationalization (i18n) will refer to the core feature used to allow users to localize (l10n) their documentation.

Therefore **MkDocs use the i18n term to refer to the fact that it supports theme text and dialogs localization (just like Jinja).**
The `mkdocs-static-i18n` plugin allows you to support multiple languages of your documentation by adding localized versions of your files to your existing documentation pages.

It also allows you to **build and serve localized versions of any file extension** and automatically display localized images, medias and assets from your Markdown sources.

Who's using it?

- spaceship-prompt / spaceship-prompt
- aws / copilot-cli
- mastercomfig / mastercomfig
- ManticoreGamesInc / platform-documentation
- index-py / index.py
- ...

Github Pages

Websites for you and your projects **hosted directly from your GitHub repository**. Just edit, push, and your changes are live.

You get one site per GitHub account and organization, and unlimited project sites.

Free **HTTPS** with support for **custom domain names**.

Soft limitations:

- 1GB website
- 100GB traffic bandwidth per month
- 10 builds (deploy) per hour

GitHub Pages is not intended for or allowed to be used as a free web hosting service to run your online business, e-commerce site, or any other website that is primarily directed at either facilitating commercial transactions or providing commercial software as a service (SaaS).
Enough introductions, let’s see some code YAML!

What we need:

- A fork of https://github.com/ultrabug/ep2021
- A Python3 virtual environment

```bash
~/ep2021 $ python3 -m venv ep2021_venv
~/ep2021 $ source ep2021_venv/bin/activate
(ep2021_venv) ~/ep2021 $ python --version
Python 3.9.5
```

Web version of this presentation: https://ultrabug.fr/ep2021
What’s required and why?

**mkdocs >= 1.2.2**

We do not need `mkdocs[i18n]` which adds support for built-in themes localization since `mkdocs-material` supports localization on its own.

**mkdocs-material >= 7.1.11**

We want a modern looking, responsive and highly customizable theme with built-in support for a language switcher.

**mkdocs-static-i18n >= 0.18**

Allows to localize our content pages by creating a `<language>` prefixed version of any file to localize it automatically.
Localized MkDocs to Github

Workflow:

1. Initialize our project
   
   `mkdocs new .`

2. Build and serve locally, open in browser
   
   `mkdocs serve`

3. Modify content, add assets and localize (translate) our website

4. Local website is auto-refreshed on browser, we preview every modification we make live

5. When done, commit and push changes

6. Deploy to Github Pages

   `mkdocs gh-deploy`

7. It’s online!
Initialize our MkDocs project

**mkdocs new .**

Will create:

- an initial `mkdocs.yml` configuration file
- a `docs/index.md` documentation home page
Preview our website locally

`mkdocs serve`

Will **build** our website and **run a local web server** so we can **preview** it directly on our browser.

Open this URL on your browser: `http://127.0.0.1:8000/`
Changes trigger a refresh

`mkdocs serve`

Any change we make will trigger a rebuild and refresh our page in our browser!
Switch to mkdocs-material

```
theme.name = material
```

Instantaneous switch from the default `mkdocs` theme to the beautiful and feature rich `material` theme.
Switch to EuroPython colors

theme.palette.primary = green

The material theme is highly customizable, let’s use EuroPython’s green color palette.
Add images/assets

The docs/ folder is where we organize our content pages and assets.

Let’s modify the home page and add the EuroPython logo.
mkdocs-static-i18n

Will create `/<language>/` versions of our website!

Automatically configures:

- search plugin language and localized content indexation
- material theme language
- material theme language switcher in the header
Localizing our content

Let’s translate and localize our website!

We localize the versions of our pages and assets by suffixing them with `<language>.<extension>`

index.md  index.en.md  +  index.fr.md

rename English version (default language is “en”)

create French version
Localizing our content

# EuroPython 2021 Home Page

![europython 2021 logo](assets/ep2021-logo.png)

## Commands
* `mkdocs new [dir-name]` - Create a new project.
* `mkdocs serve` - Start the live-reloading docs server.
* `mkdocs build` - Build the documentation site.
* `mkdocs -h` - Print help message and exit.

## Project layout

mkdocs.yml  # The configuration file.
docs/
  index.en.md  # The documentation homepage in English.
  index.fr.md  # The documentation homepage in French.
  ...  # Other markdown pages, images and other files.

# EuroPython 2021 Page d'accueil

![europython 2021 logo](assets/ep2021-logo.png)

## Commandes
* `mkdocs new [dir-name]` - Créer un nouveau projet.
* `mkdocs serve` - Démarrer un serveur avec rafraîchissement automatique.
* `mkdocs build` - Générer le site de documentation.
* `mkdocs -h` - Afficher l'aide et s'arrêter.

## Arborescence du projet

mkdocs.yml  # Le fichier de configuration.
docs/
  index.en.md  # La page d'accueil en Anglais.
  index.fr.md  # La page d'accueil en Français.
  ...  # Other markdown pages, images and other files.
Automatic MD localization

Focus on translating, not referencing localized versions of your pages and assets!

Both `index.en.md` and `index.fr.md` refer to `ep2021-logo.png` in their Markdown source:

![europython 2021 logo](assets/ep2021-logo.png)

ep2021-logo.png

and `/en/

ep2021-logo.en.png

and `/fr/

ep2021-logo.fr.png

English version (default language is “en”)
Automatic MD localization

# EuroPython 2021 Home Page

Will be localized automatically!

## Commands

* `mkdocs new [dir-name]` - Create a new project.
* `mkdocs serve` - Start the live-reloading docs server.
* `mkdocs build` - Build the documentation site.
* `mkdocs -h` - Print help message and exit.

## Project layout

```
#docs/
index.en.md  # The documentation homepage in English.
index.fr.md  # The documentation homepage in French.
...  # Other markdown pages, images and other files.
```

# EuroPython 2021 Page d'accueil

Will be localized automatically!

## Commandes

* `mkdocs new [dir-name]` - Créer un nouveau projet.
* `mkdocs serve` - Démarrer un serveur avec rafraîchissement automatique.
* `mkdocs build` - Générer le site de documentation.
* `mkdocs -h` - Afficher l'aide et s'arrêter.

## Arborescence du projet

```
#docs/
mkdocs.yml  # Le fichier de configuration.
index.en.md  # La page d'accueil en Anglais.
index.fr.md  # La page d'accueil en Français.
...  # Other markdown pages, images and other files.
```
mkdocs-static-i18n automatic MD localization demo
Push our work to Github

```bash
$ git add .
$ git commit -m "mkdocs localized website"
$ git push --set-upstream origin main
```

We’re done, let’s commit and push our content to our Github repository.
Build and deploy to Github

`mkdocs gh-deploy`

Build our multi-language website

Copy `site` dir to the `gh-pages` branch

Push the `gh-pages` branch to Github

```
~/ep2021 $ python3 -m venv ep2021_venv
~/ep2021 $ source ep2021_venv/bin/activate
(ep2021_venv) ~/ep2021 $ pip install -r requirements.txt

(ep2021_venv) ~/ep2021 $ git checkout localized
Switched to branch 'localized'

(ep2021_venv) ~/ep2021 $ mkdocs gh-deploy
INFO - Adding 'fr' to the 'plugins.search.lang' option
INFO - Cleaning site directory
INFO - Building documentation to directory: /home/alexys/github/ep2021/site
INFO - Building en documentation
INFO - Building fr documentation
INFO - Documentation built in 8.76 seconds
INFO - Copying '/home/alexys/github/ep2021/site' to 'gh-pages' branch and pushing to GitHub.
...
To github.com:ultrabug/ep2021.git
  * [new branch]  gh-pages -> gh-pages
INFO - Your documentation should shortly be available at:
https://ultrabug.github.io/ep2021/
```
Github Pages is automatically configured

GitHub Pages

GitHub Pages is designed to host your personal, organization, or project pages from a GitHub repository.

- Your site is published at https://ultrabug.github.io/ep2021/

Source

Your GitHub Pages site is currently being built from the gh-pages branch. Learn more.

- Branch: gh-pages
- / (root)
- Save

Custom domain

Custom domains allow you to serve your site from a domain other than ultrabug.github.io. Learn more.

- Save
- Remove

Enforce HTTPS

- Required for your site because you are using the default domain (ultrabug.github.io)

HTTPS provides a layer of encryption that prevents others from snooping on or tampering with traffic to your site. When HTTPS is enforced, your site will only be served over HTTPS. Learn more.
WoW our multi-language MkDocs website is online!

https://ultrabug.github.io/ep2021/
Let’s use **Github Actions** to run **mkdocs gh-deploy** for us **automatically when we push** new commits!

```yaml
name: gh-deploy
on:
  push:
    branches:
      - main

jobs:
  build:
    name: MkDocs Github Pages automatic deployment
    runs-on: ubuntu-latest
    steps:
      - name: Checkout main
        uses: actions/checkout@v2
      - name: Set up Python 3.9
        uses: actions/setup-python@v2
        with:
          python-version: '3.9'
      - name: Install requirements
        run:
          - python -m pip install --upgrade pip
          - pip install -r requirements.txt
      - name: MkDocs gh-deploy
        run: |
          git pull
          mkdocs gh-deploy
```
Some useful and awesome MkDocs plugins

MkDocs Wiki’s Plugins list (community based)

➢  https://github.com/mkdocs/mkdocs/wiki/MkDocs-Plugins

Plugin to handle URL redirections

➢  https://github.com/datarobot/mkdocs-redirects

Plugin to handle navigation pages ordering in a very flexible way

➢  https://github.com/lukasgeiter/mkdocs-awesome-pages-plugin

Plugin to hook your own functions without having to create a MkDocs plugin

➢  https://github.com/aklajnert/mkdocs-simple-hooks
Some useful and awesome Markdown extensions

```markdown
markdown_extensions:
- attr_list
- admonition
- toc:
  - permalink: true
- pymdownx.emoji:
  - emoji_index: !!python/name:materialx.emoji.twemoji
  - emoji_generator: !!python/name:materialx.emoji.to_svg
- pymdownx.tasklist:
  - custom_checkbox: true
```

![img](photo.jpg)
git checkout extensions

Check out the sample `mkdocs.yml` available on the "extensions" branch!
Thank you for attending!

@ultrabug (twitter, github, gitlab, IRC OFTC/Libera.chat...)

Repository: https://github.com/ultrabug/ep2021

Web version of this presentation: https://ultrabug.fr/ep2021

❤️ Special thanks to:

mkdocs: @waylan @oprypin
mkdocs-material: @squidfunk
mkdocs-static-i18n: @Caetan95 @Stanzilla @adamkusmirek
pymdown-extensions: @facelessuser