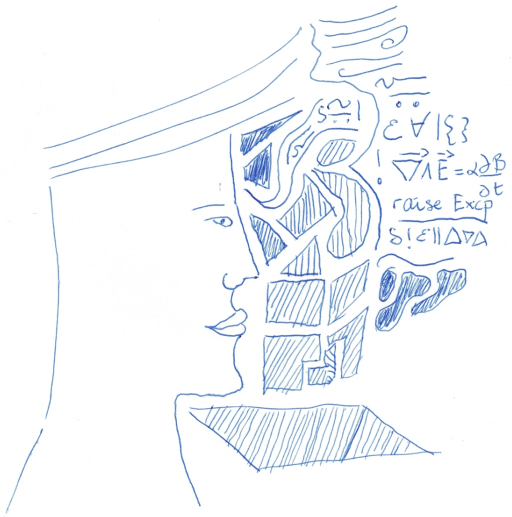
SCAM Manual

jul

Version : 202503



Contents

Synopsis 2

Installation 3

Walkthrough: writing the aide 5
- Creating your first post 5
- Attaching a content to an entry. 7
- Accessing the text 7
- Developping your first comment and setting your book title 8

Visualizing your document 9
- HTML output 9
- PDF 10

Rinse and repeat 11

Playing with the help example 12

Liste des liens 13

Synopsis

> To look smart you either say it in latin or write in LaTeX and add the Naviers Stocke equation
>
> $$\frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot \vec{j} = 0$$
>
> to look smarter

But most of us ain't smart enough to use LaTeX, at most we can use markdown an easy to learn text renderer.

So here is my solution to be a professional scamer : this is a front end to a pandoc toolchain based on mind mapping for structuring the thoughts with a real time rendering of the markdown and quite a few tricks.

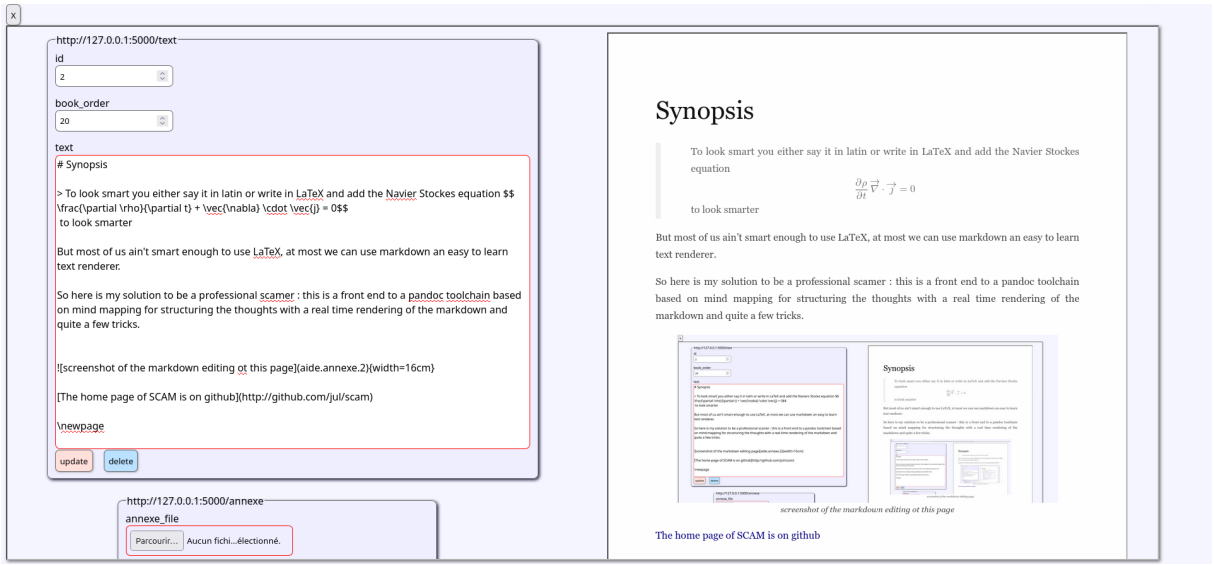


Figure 1: screenshot of the markdown editing ot this page

The home page of SCAM is on github

Installation

I am too lazy to write an install script, make a debian package, or a full pip requirements solution because I need binary installs.

As a result I resorted to the solution of the lazy man which source tells you all you need : making it a docker file.

```
FROM debian
ENV LANG C.UTF-8
RUN mkdir -p /usr/share/man/man1 && mkdir -p /usr/share/man/man7
RUN apt-get update && apt-get -y dist-upgrade \
    && rm -rf /var/lib/apt/lists/*
RUN apt-get update && apt-get -y --no-install-recommends install \
    python3 python3-pip python3-venv python3-setuptools \
    python3-sqlalchemy texlive pandoc graphviz virtualenv \
    python3-magic sqlite3 texlive-xetex texlive-latex-extra \
    texlive-fonts-recommended texlive-lang-french lmodern

RUN useradd scam -d /app --uid 1000 -m -s /bin/bash
COPY --chown=scam . /app
WORKDIR /scam
RUN mkdir /scam/assets /venv
RUN chown -R scam:scam .
COPY  . .
RUN virtualenv --system-site-packages /venv
RUN . /venv/bin/activate
COPY requirements.full.txt .
ENV PYTHONPATH=/venv/bin
RUN /venv/bin/python -m pip install --no-cache-dir \
    --disable-pip-version-check -r requirements.full.txt
EXPOSE 5000
USER scam
CMD . /venv/bin/activate && cd /scam && DB=${db:-scam} /venv/bin/python /app/scam.py
```

with the following requirements :

```
archery
pandoc-include
dateutils
multipart
filelock
```

```
Mako
pandocfilters
panflute
passlib
python-dateutil
SQLAlchemy>=2
SQLAlchemy-Utils
time-uuid
```

to use it I recommend the side car technique wich can be used this way so that you can access the assets dir which contain the book :

```
docker build -t scam .
docker run -i -t -e db=bookname \
    --mount type=bind,src=.,dst=/scam \
    -p5000:5000  scam
firefox http://127.0.0.1:5000
```

Walkthrough: writing the aide

Creating your first post

The landing page give you one option : POST :D

Figure 2: Graph interface : time to create your first post

On click you should see this and be able to fill the value:

Figure 3: First post

and then click create

Attaching a content to an entry.

I chose the policy that one micro item is related to one and only one attachment of the embedable kind you want.

The annexe widget below is used to load and delete annexes (attached files) that will be stored in the database.

These items will be available as pictures in the markdown editor by the given name on the top.

Figure 4: by clicking on the « attach » button you can attach a content to the post it entry

Accessing the text

To access the post entry you click on the node/post and it will open a modal window in which the text button is accessible for editing your book text

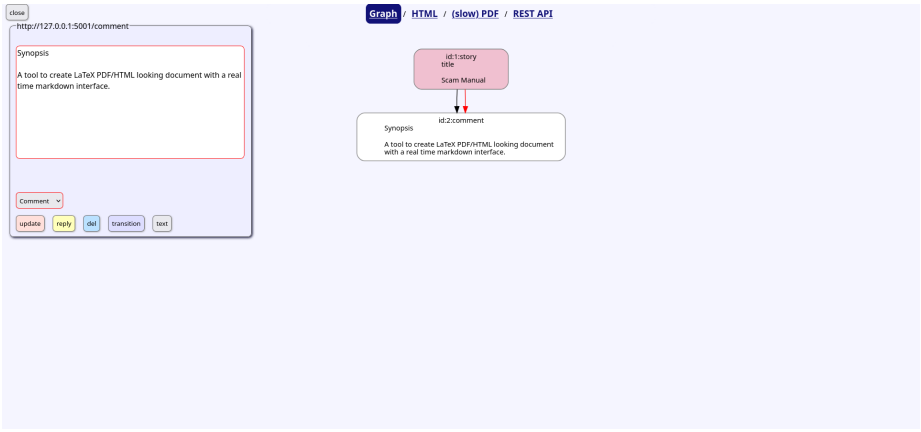


Figure 5: Once you click on a node in the graph the post editing interface appear

Developping your first comment and setting your book title

First comment is specific in the sense it is also used for the title. With pandoc you can add metada used for LaTeX.

The markdown extension useds here is the pandoc one

Here is a typical Pandoc flavoured Markdown entry to setup the LaTeX settings here with the french settings :

```
% TITLE
% AUTHOR
% DATE \
  \
  \ ![](aide.annexe.1){width=15cm}

---
header-includes:
 - \usepackage[french]{babel}
 - \usepackage{hyperref}
 - \definecolor{myblue}{rgb}{0.28, 0.24, 0.48}
 - \hypersetup{colorlinks=true, allcolors=myblue}
 - \let\tmp\oddsidemargin
 - \let\oddsidemargin\evensidemargin
 - \let\evensidemargin\tmp
  - \reversemarginpar
 ---
```

Your real time markdown input is definitely confused but that's fine. It is a quirk :D

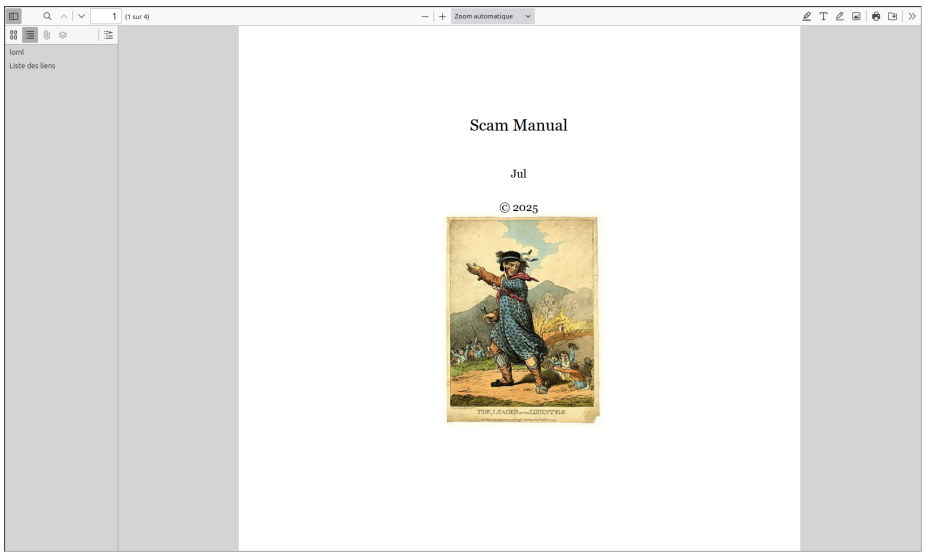


Figure 6: screenshot of the rendered PDF with the title

Visualizing your document

Now that you have rinced and repeated a few time the post entry/text process you may want to check the output.

There are 2 main output : HTML and PDF.

HTML output

To see the result

it's now time to visit the HTML rendering URL. You should have a side by side view of the generated standalone HTML and the generated PDF.

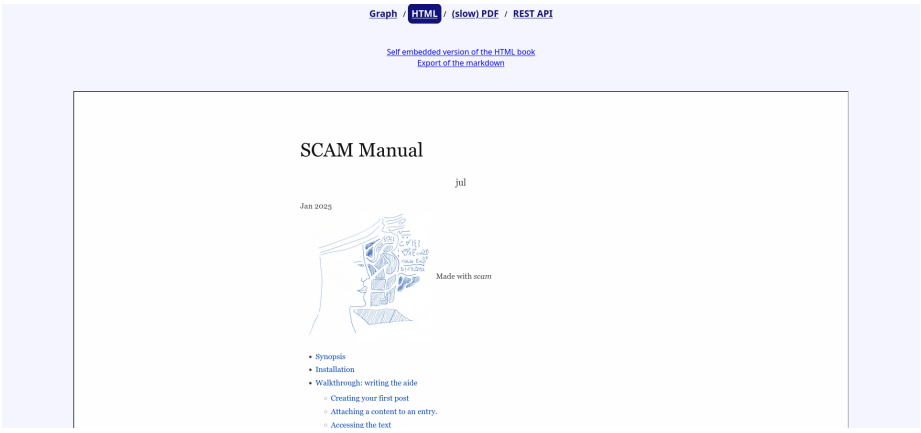


Figure 7: book rendering

You notice that there is a nice self embedded HTML link. This includes CSS and pictures inside the document for serving the document as a single file.

PDF

The PDF renderer accessible from the menu will give you the PDF rendering.

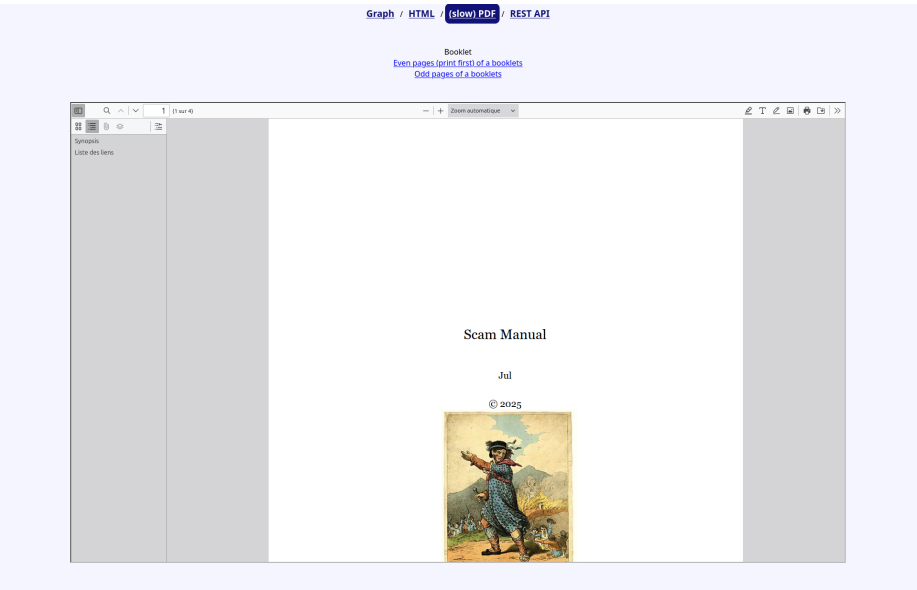


Figure 8: PDF view of the document

Rinse and repeat

Ather a few more entries that are boring because very repitive if you consult the graph URL you should have now more entries.

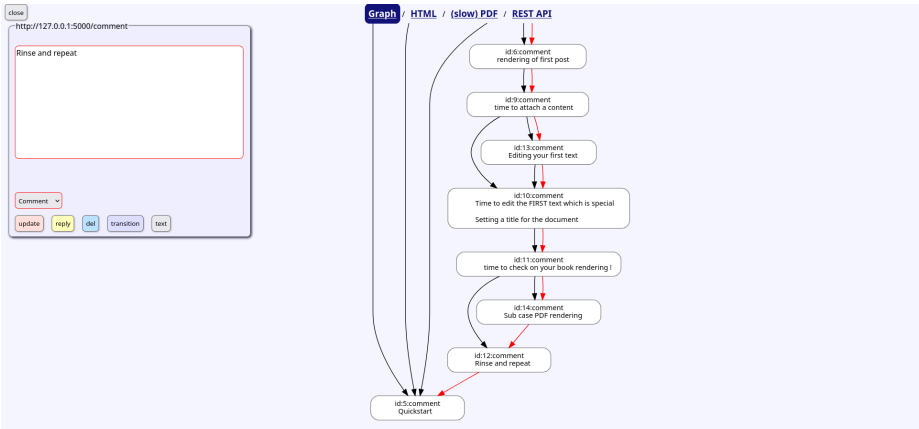


Figure 9: Your graph should now expand itself as the book

The « book order » is the red lines, they follow the ascending id order but can be overriden with the book_order rank available in the text view.

Playing with the help example

This book is available in the repository as a sqlite database.

To try it :

```
docker run -i -t -e db=aide --mount type=bind,src=.,dst=/scam \
    -p5000:5000 --user  1000:1000  scam
firefox http://127.0.0.1:5000
```

Liste des liens

http://127.0.0.1:5000/ The landing page
https://pandoc.org/MANUAL.html#pandocs-markdown markdown exten-
sion useds here is the pandoc one
http://127.0.0.1:5000/book it's now time to visit the HTML rendering URL
http://127.0.0.1:500/pdf PDF renderer
http://127.0.0.1:5000/svg you consult the graph URL
http://github.com/jul/scam The home page of SCAM is on github