Docs for public software projects

Ankur Sinha

29/10/2020

Documentation?

Docs for different target audiences: dev docs

- In-source comments: help to understand the source code,
- Change log: what changed and when (version control).
- Standard Operating Procedures (SOPs): ensure no single point of contact/failure (SPOC/F):
  - API/ABI versioning and software release SOP,
  - Continuous integration (CI) SOP,
  - Deployment SOP,
  - Contribution guidelines: pull request workflow etc.

In-source comments: not for users

```
/*
* Dear Maintainer
*
* Once you are done trying to 'optimize' this routine,
* and you have realized what a terrible mistake that was,
* please increment the following counter as a warning
* to the next guy.
*
* total_hours_wasted_here = 73
*/
```

[1] Top 45 Best Comments In Source Code I Ever Encountered

Change logs/commit logs: not for users either

[2] xkcd: Git Commit

[3] How to Write a Git Commit Message

Docs for different target audiences: user docs

- Tools: how to use the tool, options, features, extensions . . .
- Libraries: how to use the API,
- Example code snippets, guides.

Documentation for users: separate from development

- **Must** clearly define what the tool (software) does,
- **Must** define entry-points for users to get started with the software,
- **Must** provide a high level overview of concepts needed to use the software,
- **Should** provide references to detailed definitions of these concepts,
- **Must** list channels of communication etc. for users,

Documentation for users: separate from development

- **Must** clearly define what the tool (software) does,
- **Must** define entry-points for users to get started with the software,
- **Must** provide a high level overview of concepts needed to use the software,
- **Should** provide references to detailed definitions of these concepts,
- **Must** list channels of communication etc. for users,
- **May** be generated from comments in the source code but **must not** be written like developer documentation.

Publishing documentation: software

Documentation tools: translators

Docs sources $\rightarrow$ documentation tool $\rightarrow$ HTML/PDF/...

Today: Sphinx, Jupyter-book.

Sphinx: Python default

- Inputs sources:
  - reStructuredText (default),
  - Markdown (CommonMark *flavour*[5])
  - MyST (Markdown + reStructuredText)[6]
- Output formats:
  - HTML (multi-page and single-page),
  - LaTeX for PDFs, ePub, Texinfo, Man pages, . . .

[4] Overview — Sphinx 4.0.0+ documentation

[5] The original Markdown specification is not unambiguous, and so multiple flavours of Markdown have cropped up over the years. CommonMark is one attempt at standardisation. GitHub Flavoured Markdown is another that GitHub created.

[6] MyST - Markedly Structured Text

Sphinx example: OCNS Software SIG documentation

Demo!

Jupyter-book: shiny new Jupyter based system

- Input sources:
  - Jupyter Markdown,
  - Jupyter notebooks: Python, Julia, Ruby, Haskell ...,
  - MyST,
  - reStructuredText.
- Output formats:
  - HTML (multi-page and single-page),
  - PDF.
- Interactive pages!
  - Binder, JupyterHub, Google Colab, ThebeLab[8].

[7] Books with Jupyter

[8] Embedded: so no need to leave the page.

Jupyter-book example: NeuroML documentation

Demo!

Lab website updates? Time?

www.silverlab.org