Technical Writing Assignment

Brandi McPherson

October 9, 2024

Task: Analyze, reorganize, and rewrite the Visibility Calculators guide, focusing on improving clarity, structure, and usability.

1 Analysis Paragraph

Overall, this guide is moderately easy to digest but is missing a few core components I would prefer to see. For strengths, charts and code blocks are always a great addition and useful method to break up the monotony of a page itself. Links to other pages containing relevant objects are another clear strength. For weaknesses, there are inconsistencies where brief summaries of items are missing. "Note" sections are great to have, but I find the green text does not stand out enough to grab my attention. Grammatically, the writer(s) switches between 2nd and 3rd person, referring to the reader as both "you" and "the user." I have a bias towards 2nd person but overall want to see consistency throughout a document. I would leave most of the guide structure intact, with the exception of adding or moving a few sections for easier flow and creating a table of contents for better navigation. The content itself was thorough, but missing an initial narrative that ties `VisibilitySegment` and the `VisibilityCalculator` together.

2 Rewriting Task

Please view the modified documentation on my GitHub[1].

3 Justification Paragraph

Starting with the components I preserved, all code blocks as well as the "Refraction Table" remained intact[2]. Code blocks are not only important as examples but prevent a user from losing focus in walls of text. I preserved hyperlinks to relevant pages; these prevent users from searching for dependencies outside the page[3]. Structurally, the original page could only be navigated by scrolling or selecting a header link. I added a table of contents to act as an overview and provide easy access to specific topics. For users that are only searching for examples, I included dedicated "Examples" and "Scripts" sections[4]. I moved "References" to the bottom of the page to not break the flow[5] To better grab the user's attention, I moved all notes to dedicated boxes and labeled one as a "warning". I provide extra attention to changes that may impact a user's existing simulation. Grammatically, there were a handful of sentences I rewrote for clarity and conciseness. To address "knowledge gaps", I included additional summaries for major topics as well as any required inputs for class methods. Most importantly, in order to portray a unified voice, I rewrote all text to address the reader in 2nd person. This prevents using passive voice too frequently and creates a "connection" between the reader and narrator. The modifications I made are both visual and contextual; good documentation must consider both aspects. Overall, I did not recommend drastic changes for this guide. When rewriting documentation, I want to reuse as much as I can; however, this is not always the case and more dramatic solutions are recommended.

[1] The guide was generated using the Python Sphinx package. Sphinx generates HTML and PDF (via LaTeX) from reStructuredText files. I am using GitHub Actions to deploy the HTML pages

[2] I didn't fight with the table formatting for this assignment; usually I would set the table as the page width

[3] Additionally included hyperlinks to the class methods to view input arguments and types

[4] The "Examples" section was previously a dialogue box at the top of the page. The "Scripts" section combines all code snippets into one block for better readability

[5] I would usually hyperlink the references from the table to the footnotes, but for the purposes of this assignment, I did not.