Guidelines for <Department> Docs (v2.0)

This document is intended to help an SME who's been tasked with drafting a document understand the basic principles of writing good documentation and to provide an introduction to how to do specific tasks like creating headings, adding and annotating graphics, tag your doc appropriately, etc.

Document Goals

As an SME drafting technical documentation for your customers, your primary goal is to get the customer (the reader) the information they need to succeed in their job. The information you share must be:

Correct

Incorrect documents are worse than no documents at all.

You were asked to be the SME responsible for drafting this document. That does not mean you have to be the expert on every aspect of the task – but it does mean you need to consult an expert in any area you're not knowledgeable about. When it doubt, check. When you've completed your draft, see if you can ask a colleague to check it over for you. If the draft is correct in the beginning, you won't have to spend much if any time on revisions and corrections.

Simple

Simple, concise documents make it easy for your customer. If your customer has to struggle to understand the information, at best they will be delayed. At worst, they will misunderstand what you said and fail at the task.

Use a consistent structure for all docs of a given type, so the reader knows what kinds of information will be available where.

- Use informative headings and subheadings.
- For longer docs, break the information into separate tabs (as in the Tech Manual template) or provide a Table of Contents that helps the reader navigate through the document.
- Include any references the reader may need in order to understand what you've written – remember, often the reader will have far less experience than you do. Linking to glossaries, "about" articles, or other resources can make a big difference. In addition, linking out to additional information keeps your document pared down to the actual essentials.

Shorter is almost always better.

Clear

When you read what you've written, try to notice anything that may be unclear – or ask someone who doesn't understand the material yet to read it and let you know what they find confusing. Something as simple making it clear what a pronoun (in this example, "it") is referencing can make a huge difference.

Consider:

- "if it's tall enough, plant the sapling and secure it to the support" – *this could be confusing, especially for someone who's not used to planting trees!*

- "if the support is tall enough, plant the sapling next to it and secure it to the support" – *ah, the support has to be tall enough to provide the support the sapling needs… How tall does it need to be?*

- "If the sapling is tall enough, plant it next to the support and secure it" – *oh, there's a minimum height to plant a sapling. What is that? How can the reader know? Is it the same for all kinds of trees?*

If you use a consistent structure, it will become easier for you to create the documents needed because you'll be more familiar with what you're doing. And if the draft document is clear, you won't have to spend much time explaining what you meant to your colleagues or to your technical writer/editor/reviewer.

Complete

The customer should not have to go searching for additional information they need to complete their task.

If there's information they may need that's beyond the scope of your document, provide links to that material so it's easy for the person reading your document to get everything they need.

It's easy for an expert to take things for granted that a new user may not actually know they need to do. If you were giving driving instructions, would you begin with "start the car"? – or would you think to include "make sure you have the car key" and "put the car key in the ignition, put the car in park or in neutral, step on the brake, and turn the key"? (Or even "are you going to be driving a manual-shift car or an automatic?")

Make sure your doc either explains or links out to any information the reader will need in order to complete their task, even if it's something that you would take for granted.

Basic Principles

People who are looking for documentation are generally already in the middle of a task. They are not going to take the time to read carefully or figure out a paragraph that doesn't make sense the first time through. It's up to the author to make it as simple as possible for their reader to understand exactly what they need to know to complete their task correctly and go on with their day. There are a few basic principles that will help with this.

"Tell Them Three Times"

Also known as "Tell them what you're going to tell them. Tell them. Tell them what you told them."MoreThat is, in any simple piece of writing, you should have a sentence or two that says exactly what the doc is about. Then there's the body of the doc, where you deliver on what you promised in that one- or two-sentence beginning. And at the end, to help them remember the content, you should give a very brief summary of what you just said. For example, this document begins with --

> *This document is intended to help an SME who's been tasked with drafting a document understand the basic principles of writing good documentation and to provide an introduction to how to do specific tasks like creating headings, adding and annotating graphics, tag your doc appropriately, etc.*

"Keep It Simple"

This is sometimes seen as "Keep it Short and Simple" or "…Sweet and Simple," but "Keep It Simple" is the simplest way to say it. The simpler and more straightforward you keep your structure – and your sentences – the easier it is for the reader to understand what you're saying.

"Love the Spaces"

The more complex your topic is the more important it is to keep your paragraphs short, use numbered or bulleted lists, provide screen captures or diagrams, and make sure the screen your reader is going to be looking at is clean, simple, and not crowded.

Specific Guidelines

Structuring Your Content

Just as you need to have a structure in mind when you're building something – a floor plan for a building, a pattern for handicrafts, or the architecture for software – you need to have a structure in mind when you're writing documentation. The headings can be used to reflect that structure – in a **How-to doc**, it's a simple structure with just an Introduction section and a Solution section, where in a **Tech Manual** there may be more content, and therefore more structure needed. The structure of a document helps the reader understand what information they can find in the document and where to look for it – it lowers the bar for the reader to understand what you're trying to convey.

The goal of a **How-to doc** is to help the user solve a specific problem or accomplish a specific task. Often, the user is in the middle of an activity when they realize there's part of the process that they need help with, and they want to get that help and get back to their work as simply and quickly as they can. In other cases, the user has run into a problem and something isn't functioning as expected; again, they want to solve the specific problem in front of them and get back to the work in hand.

The goal of a **Tech Manual** is broader: it is intended to help someone understand how a given process or tool functions from beginning to end. Therefore, it needs to include everything the user needs to understand the process or tool and use it effectively. This may include any or all of pre-requisites and dependencies, tool installation and setup, process flow or tool use, troubleshooting, and links to any additional resources that may be useful. Someone who is turning to a multi-page doc is still probably not reading it from beginning to end, but they will reference it for a fuller context and more detail than a FAQ can provide.

Structuring a FAQ/How-to Doc

For a FAQ/How-to document, you only need to worry about creating an introduction and then describing the solution, so it's pretty straightforward. If you wanted an outline for your FAQ, it might look something like this:

```
I.    Introduction

      A. Why is this FAQ needed?

      B. Who is the intended audience for this FAQ?

      C. What will this FAQ help the reader accomplish? ("Tell them
         what you're going to tell them.")

      D. What tool or tools are relevant?

      E. What must the reader have in order to be able to complete
         this process? (e.g. access to a tool)

II.   Solution

       A. Steps

          1. Step One

          2. Step Two (etc.)

       B. Links to additional information or more help
```

Not every FAQ will have content for all these elements – this is only an example.

Structuring a Multi-Page Doc

The template for multi-page documents provides a tabbed structure that includes tabs across the top of the page for:

- **Introduction or Overview:** This tab introduces the topic and provides any context the reader will need to understand how and why to use the tool or process.
- **Getting Started:** This tab covers any pre-requisites (hardware requirements, permissions, etc.) and either describes or links to any installation instructions.Installation instructions and similar content should appear in only one document and that document should be referenced wherever else those instructions may be needed. Having multiple versions of a process in various documents leads to confusion and to difficulty maintaining the documents.
- **Documentation:** This tab covers every aspect of the process or tool use (after meeting the pre-requisites/managing installation), from initial set-up onward.
- **Troubleshooting:** This tab should point the users to how to resolve common issues and where to find any additional information they may need to resolve whatever problems they've discovered.
- **Appendices:** This tab should include any useful reference information and links to further information about the tool.

Required Tabs

The only tabs that are absolutely required are "Overview" and "Documentation," but in many cases all tabs will prove to be useful.

Multi-page docs are more diverse than FAQs or How-To documents, and determining the right structure or organization can therefore be more difficult. However, the basic principles are the same as was described above: "Tell 'em what you're going to tell 'em. Tell 'em. Tell 'em what you told 'em."

- Use the "Introduction" section to give the reader the context they need to understand the material –
    - What are they doing, why are they doing it, and what will the desired outcome be?
    - What are the major parts of the content they will need to understand?
- Each major part of the content should have its own section. If it's complex, it may benefit from having its own introduction, too.

For example, an automobile repair manual will probably have a general introduction to the vehicle followed by major sections for each major component of the car – engine, transmission, brakes, etc. Each of those sections will include an overview or introduction before addressing the details of construction, service, and repair.

Quick Tips:

Creating Titles

Titles should describe the document as simply and clearly as possible. However, do not sacrifice clarity for simplicity. Titles are key to effective searchability.

If the document is about managing settings for a given tool to avoid having your keyboard unexpectedly changed from QUERTY to Dvorak, for instance:

Consider This	**NOT This**
“Restoring Your QWERTY Keyboard in Your <Tool> Settings”	“Fixing Your Keyboard”

Short titles are great – but not at the price of the reader understanding right up front what information is going to be in this document.

Creating Headings

Headings help your reader navigate your document, and can also be used to create a table of contents or even link to a specific point in your document. To be useful, headings should be simple and should be descriptive of the content found under them.

All the headings in a given section of the document should follow the same structure – for instance, in this document, all the headings under "Specific Guidelines" begin with an action word – creating, outlining, choosing, using, finalizing.

Here are examples of good and not-so-good ways to create headings:

Good	**Good**	**Not So Good (Inconsistent)**	**Not So Good (Unclear)**
1. Getting Started with <Tool>	1. <Tool> Setup	1. Setting Up <Tool>	1. <Tool>
2. Using <Tool>	2. <Tool> Functionality	2. <Tool> Functionality	2. Getting <Thing>
3. Updating <Tool>	3. <Tool> Updates	3. Updates	3. Check <Vendor Site>
4. Deleting <Tool>	4. <Tool> Deletion	4. Removing <Tool>	4. <no heading, continues from above>

Note that in the How-to documents, you should not need to use any heading levels other than "Heading 1," but in the Tech Manual docs you may need to use additional levels. In both the document templates, your initial top-level headings are automatically created.

Creating Lists

You can easily create bulleted or numbered lists in Confluence by clicking on the appropriate "lists" icon in the formatting toolbar. Be aware, however, that Confluence's list functions are limited, and do not reflect the full functionality found in many other tools that enable lists. Use lists at your discretion, but do not expect to be able to create complicated multi-level or multi-type lists in Confluence.

- Use a numbered list when the order of operations presented is vital. Use a bulleted list when you want to stress the parallelism of a number of options, elements, rules, or instructions that need not be presented or performed in a particular order.
- Introductory punctuation: Precede all lists with colons, whether the sentence before the colon is a complete thought or a partial thought (exception to The Chicago Manual of Style).
- Capitalization: All list items should start with a capital letter unless the word is a product name that begins with a lowercase letter or, in developer materials, is a computer literal that begins with a lowercase letter.
- Punctuation within a list: List items that are fragments or that complete the thought started by the main clause should not end with a period; list items that are complete sentences should end with a period. All list items in a particular list should be of one type or the other type – they should not be mixed.
- Parallelism: Within a single list, all bulleted items should be parallel.
- A regular sentence broken into a list: This type of list emphasizes the parts of a series. The syntax of the sentence is unbroken.
- A simple list: The main clause is an independent clause and each bulleted item is a sentence fragment.
- A complex list: The main clause is an independent clause and each bulleted item is a complete sentence.

Choosing Your Words

*Use the simplest word that conveys your meaning***.**

In technical docs, the longest word or the most jargon-filled word are not necessarily the best choice – your goal is to be understood, not to impress the reader. In the Avoid Jargon section of the plainlanguage.gov site, they use the following example:

> […When w]e say not to use jargon, we're not advocating leaving out necessary technical terms, but we are saying to make sure your language is as clear as possible. For example, there may not be another correct way to refer to a "brinulator valve control ring." But that doesn't prevent you from saying "tighten the brinulator valve control ring securely" instead of "Apply sufficient torque to the brinulator valve control ring to ensure that the control ring assembly is securely attached to the terminal such that loosening cannot occur under normal conditions." The first is a necessary use of a technical term. The second is jargon.

Choose the simplest words and phrases that will still communicate your meaning accurately. Simple words and short sentences are your friends.

Conclusion

Documentation does not have to be difficult to write. Remember to keep it correct, simple, clear, and complete; follow the existing style guides or templates; get a peer to review your document for accuracy, and if possible ask your team's writer to review it for structure, flow, and language; then you should be in good shape!