Kindle Technical Documentation SOP

Contents

1. Introduction

This wiki documents our team's best practices for requesting and creating technical documentation. Our goal is to create a consistently high-level experience for the customers of our software user guides and other technical documents.

2. Requesting Documentation

2.1. Overview

This section of our wiki describes the process for requesting new or revised technical documentation from Tricia Spicer Bekey via SIM. Some things to be aware of before you file your request:

- During the process, we will ask you to review drafts and we will have questions. We ask for review feedback within 3-5 business days; returning feedback faster accelerates the schedule. **If you will be unavailable for reviews and/or questions, please designate someone else to do this or reconsider your timeline.**
- If you are requesting software documentation, be ready to provide an estimated date of a **feature-complete (FC) build** (for first draft) and an estimated date of a **zero bug bounce (ZBB) build** (for final updates to text and screenshots). Be aware that if these build release dates change, the documentation schedule will change to match them.

    - **Stable feature-complete (FC) build:** A feature-complete build has all of its planned or primary features implemented but is not yet final due to bugs, performance, or stability issues.
    - **Zero Bug Bounce (ZBB) Build**: The zero bug bounce build meets two criteria: (1) The rate of bug fixes is greater than the rate of bug creation. (2) There are no active bugs that are planned to be fixed before release.
- This process applies to technical documents that do not have an SOP. For the Kindle Publishing Guidelines &/or the Publish to Kindle User's Guide, see those SOPs.

2.2. Update Process

This is our current update process:

Task	Owner	Details
Request new or revised technical documentation	Amazon employees	Amazon employees request new or revised technical documentation via the Kindle Technical Documentation SIM.
Provide applicable requirements docs	Original requester	The original requester will provide applicable requirements documents for the SIM request evaluation and first draft.
Will documentation require localization?	Original requester	If your documentation will be released outside the United States, include the list of required languages in the SIM request & contact the localization team with your documentation requirements. They will provide input on the schedule.
Is this software documentation?	Original requester	If you are requesting software documentation, be ready to provide an estimated date of a **feature-complete (FC) build** (for first draft) and an estimated date of a **zero bug bounce (ZBB) build** (for final updates to text and screenshots). Be aware that if these build release dates change, the documentation schedule will change to match them. • **Feature-complete (FC) build:** A feature-complete build has all of its primary features implemented but is not yet final due to bugs, performance, or stability issues. • **Zero bug bounce (ZBB) build**: The zero bug bounce build meets two criteria: (1) The rate of bug fixes is greater than the rate of bug creation. (2) There are no active bugs that are planned to be fixed before release.
SIM request evaluated. Schedule created and shared. [SLA: 3 bus. days]	Technical writer/editor	The technical writer/editor evaluates the SIM request, creates a schedule, and shares it with the original requester (and localization, if doc will be localized).
If this is a software doc,	Original	If this is a software doc, the original requester provides a

provide a feature-complete (FC) build.	requester	**feature-complete (FC) build**, which has all of its primary features implemented but is not yet final due to bugs, performance, or stability issues. The first draft cannot be done without it.
First draft created & sent for review [SLA varies by complexity, avg. 10-15 bus. days]	Technical writer/editor	The technical writer/editor creates the first draft & sends it to the original requester for review. If this is software documentation, a feature-complete (FC) build is required for this step; if this build is not available, the first draft will be rescheduled.
First draft reviewed [SLA: 3-5 business days; returning feedback faster accelerates the schedule]	Original requester	Requester reviews doc & provides feedback. If you will be unavailable for reviews and/or questions, please designate someone else to do this or reconsider your timeline.
Second draft created & sent for review [SLA varies by complexity, avg. 3-5 bus. days]	Technical writer/editor	The technical writer/editor incorporates feedback from the first draft, creates the second draft, & sends it to the original requester for review.
Second draft reviewed [SLA: 3-5 business days; returning feedback faster accelerates the schedule]	Original requester	Requester reviews doc & provides feedback. If you will be unavailable for reviews and/or questions, please designate someone else to do this or reconsider your timeline.
If this is a software doc, provide a stable final build	Original requester	If this is a software doc, the original requester provides a **zero bug bounce (ZBB) build**, which meets two criteria: (1) The rate of bug fixes is greater than the rate of bug creation. (2) There are no active bugs that are planned to be fixed before release. The final draft cannot be done without it.
Final version created [SLA varies by complexity, avg. 3-5 bus. days]	Technical writer/editor	The technical writer/editor incorporates feedback & creates the final version with screenshots. If this is software documentation, a zero bug bounce (ZBB) build is required for this step; if the ZBB build is not available, the final version will be rescheduled. • If doc is not being localized, the technical writer/editor delivers this version to the document owner, unless the owner requests otherwise. • If doc is being localized, the technical writer/editor delivers this version to KGKM for localization.

Translations (if doc is being localized) [SLA will be provided by team]	Localization team	The technical writer/editor cuts a Remedy ticket to have our team localize the document.
Post-localization checklist [SLA: 1-2 bus. days]	Technical writer/editor	After translation, the team returns the localized Word docs to the technical writer/editor. The technical writer/editor checks the Word docs against the Post-Localization Checklist to make sure they are ready for public release & delivers the final docs to the document owner, unless the owner requests otherwise.
Special circumstances: If there are small content changes after delivery of final version (software changes, etc.).	Original requester	Although we prefer to avoid it, we understand that there are sometimes small changes required after the delivery of the final version (example: changes to UI). If this happens, provide details about the change to the technical writer/editor for rework (and localization, if doc is localized).

3. Creating Documentation

3.1. Front Matter

- Cover page of user guide should include:
  - Name of software
  - Software version number (include "Beta" if true)
  - Revision date (Important: If the software team asks for additional changes after we hand off the doc, update the revision date to keep the versioning clear.)
- Revision History: Unless the software is in beta or it is the first version of the user guide, there should be a revision history table between the cover page and the table of contents that describes which sections were changed in the update.

3.2. About Beta Software

Our current practice is to delay revision history and localization for beta software documentation because the large amount of churn inherent to the beta process makes it an ineffective use of time and money.

3.3. Quick Reference – Writing User Guides

3.3.1. Procedures

In documentation, a procedure is a description of the steps a user takes to complete a specific task. Sections that describe a procedure must have:

- A title that starts with a gerund. (**Example:** Printing Your File).

- An introductory phrase that stands on its own, does not continue from the title, and ends with a colon. (**Example:** To print your file:)
- Numbered steps that start with a capitalized word, are complete sentences, and end with a period.

Note: There are two situations where a procedure gets a bullet. When a procedure only has one step, it gets a bullet instead of the number "1." When a procedure "branches" or has several options within one procedure step, the options each get a bullet because they will not each be done in sequence.

3.3.2. Conventions

- The following items are bold: commands, dialog box options, dialog box names, folder and file names, icon names, menu names, text entry, and text displayed on the screen.
- Cascading menu sequence: Choose **Start > Settings > Control Panel**.
- Follow the capitalization of the interface.

3.3.3. Grammar

For better translations, use short, simple, easy-to-understand words and sentences. Best practice is to write in:

- **Active voice:** "A **does** B" not "B **is done** [by A]."
- **Present tense:** "The system **updates** the database." not "The system **will update** the database."
- Second person for instructions and procedures: "Click OK."
- **Third person for everything else:** "The administrator must reconfigure the system."

3.3.4. Word Choice: Actions

These terms are most commonly used for user actions in dialog boxes:

- **Click:** Use for commands, command buttons, option buttons, and options in a list, gallery, or palette. Use for mouse procedures.
- **Select and clear:** Use for check boxes.
- **Type** or **select**: Use to refer to an item that the user can either type or select in the accompanying text box. OK to use "enter" instead if there's no possibility of confusion.
- **Choose** and **select**: Use for generic procedures only, not mouse procedures. Use "choose" for commands and "select" for options.

3.3.5. Word Choice: Dialog Box Elements

- Don't differentiate between drop-down combo boxes, list boxes, etc., but do use "check box," "tab," and "slider."
- Use the label (example: **Save as type**) with "box" or "list" if needed to explain where the user should be, then direct the user to click, select, or take other action.

3.4. Style Sheet

All of our technical documents use the same styles to indicate two types of special text: code and user interface (UI) text.

Type of Text	Markup	Localization Rules	Examples
Code (HTML, CSS, XML)	**Code:** Courier New **Text:** Arial **Localized parts of code examples:** highlighted Courier New or Arial	Code tags and attributes should not be localized because the code itself is English-based, but comments, text, and some values should be localized to make our guidance as globally accessible as we can. When code examples are added or revised (as indicated with tracked changes), KGKM has asked us to highlight the parts that should be localized for clarity.	<audio id="audio_1" src="audio.mp3" controls title="Audio about ..."> <!-- If users view this eBook on a device that does not support audio, they see the following text instead --> "There is audio content at this location that is not currently supported for your device. The caption for this content is displayed below." </audio>
UI text (menus, buttons, dialog box names, all other text on screen)	bold	References to UI text must match the UI strings for the software.	Right-click the zip file, select **Extract All**, and enter the folder name as **c:\Kindlegen**.

3.5. Post-Localization Checklist

Before releasing translated technical documents to the public, we always check the following as part of best practices:

1. **Revision History Table:** Review the Revision History table (between the cover page and the table of contents) to see which sections were changed in this update. If there is no Revision History table, check to see if the guide is completely new/revised.
2. **Tracked Changes & Comments:** Make sure that all tracked changes have been accepted (not just hidden) and that all comments are deleted.
3. **Cover:** Confirm that the version number is accurate.

4. **TOC:** If any sections were added or deleted, make sure the table of contents has been updated to reflect this change. [With tracked changes off, click CTRL +A to select all text, right-click and choose **Update Field**, from the pop-up menu, choose **Update entire table** and click **OK**.]
5. **Fonts:** For each revised section:
    a. Make sure the body font family and font size matches the body style for that document. Compare to English version if unsure.
    b. Make sure any code or UI text is formatted correctly per the Style Sheet. Compare to English version if unsure.
6. **General Check:** Page through the revised sections and compare to the English version to check for any oddities, such as missing or duplicate screenshots.