Good Wiki Editing Practices

> In order to create a wiki editing account, ask for one on the Discord server.
>
> New users will require their edits get approved by an approver first before it gets shown to the general public.

Recommended reading: https://www.dokuwiki.org/tips:good_style

This page is about employing good practices when editing the wiki. Not the "how" to do something in the wiki, but the "why". None of these are strict rules, but instead guidelines to help create higher quality articles.

The Batocera Wiki is a collaborative project aimed at both introducing new users to Batocera's flow and giving extensive documentation to developers to help compiling the project. With that being said, here are some questions you should ask yourself before making an edit:

- Is the information accessible to someone with less knowledge, and if not, can other pages/sources be linked to help?
- Is the information relevant to the current page?
- Before making a new page, can the information be added to an already existing page?
- When making a new page, is the page name concise, yet precise enough for external search engines to find?
- Can the page be put into a relevant sub-category? For example, something purely related to a single system should be appended to its relevant system page.
- Is the information somewhere else already, and if so, is it in the correct place?
- Ultimately, *something* is better than **nothing**. Even if the information is not robust or high quality enough for the wiki, it's better than a complete absence of it. It can always be revisited to be amended later. The playground can be used if help is needed with certain edits, without going straight to publication.

Once that's all checked for, you can begin editing the wiki.

Syntax

DokuWiki uses its own markup syntax that's similar to but not quite markdown. You can find out the specifics on the DokuWiki syntax page. The Batocera Wiki-specific syntax can be found at the Batocera Wiki code snippets page, though really that's more just commonly used code that's interchangeable between pages like button symbols, page templates, etc.

Practicing

When starting out, you should use the playground to test things. This page is ordinarily not visible to public search engines so any changes made here are inconsequential. You can also preview your

changes before committing them with the Save button, it is good practice to do this with every edit to ensure that links and embedded images are working correctly.

If uploading images to a page in the playground, be sure to set the namespace of the image to your intended finalised namespace. Otherwise, when the page gets moved from the playground to its official namespace, all the images will not be visible to guest users!

Spelling and formatting

It's important to make sure you're using correct spelling and grammar, but for large documentations such as this wiki it's also useful to be consistent in the spelling of Batocera-specific terminology and to be consistent in the formatting of things such as buttons and menu items. It's generally more a rule-of-thumb than a strict guideline, but should you want your edits to look higher quality you should follow these conventions:

- **General spelling**
  - This wiki uses American English, so as to be approachable by the widest audience possible and to assist with the auto-translator. Spelling for words such as "color" and "bastardization" should use their American spelling. This is consistent within Batocera itself as well.
  - Level 1 title headers should be Title Cased. Remaining headers should use standard capitalization. No period marks should appear at the end of any header.
  - Unordered bullet lists and tables should have each element end without a period mark, unless the list entries are full sentences (such as this bullet list).
  - "Batocera" and "Buildroot" are proper nouns, so they should always be capitalized. For instance, this is how Batocera and Buildroot appear in the middle of a sentence. Batocera.linux is the full name of the project, and is usually used when referring to the [Github page](). It follows standard capitalization rules. For instance, this is how batocera.linux appears in the middle of a sentence.
  - When referring to specific Batocera versions, use "Batocera **v<version number>** and lower" or "Batocera **v<version number>** and higher", where the "**v<version number>**" is **bolded**. Batocera versions before **v29** should be referred to by the old versioning scheme (eg. Batocera **5.27**), whereas new versions should use the single-number versioning scheme (eg. Batocera **v29**). There is no Batocera **v28**.
  - "EmulationStation" is stylized by having capitals for both the "E" and "S". An accepted abbreviation of this is "ES", but try to use its full name especially when it's only mentioned once in the document.
  - "Wi-Fi", "Bluetooth" are stylized proper nouns, and should always be capitalized in their stylized format. For instance, that's how the [compatible Wi-Fi/Bluetooth dongle list]() handles it.
  - "RetroArch" is stylized by having capitals for both the beginning "R" and the "A" in "Arch". "RA" is not an acceptable abbreviation due to its ambiguity and should be avoided.
  - "libretro" is stylized by only using lower-case letters. An accepted abbreviation of libretro cores this "lr-<core name>", but try to use its full name especially when it's only mentioned once in the document. "lr" by itself is not an acceptable abbreviation due to ambiguity and lack of context, and should be avoided.
- **Paths and file extensions**
  - Paths to files/directories should always use path formatting. For example,

`/userdata/roms/psx/`, which is encoded as ''/userdata/roms/psx/''.

- ○ Paths to directories should end with a / and paths to a file should end with their extension (if they have one).
- ○ Certain file types should be referred to by their capitalized name. For example, the ISO file. It is also acceptable to use path notation with wildcard characters to indicate what can be replaced, such as `*.iso`.
- ○ If using a path at the end of a sentence punctuation should still be used. For example, `/userdata/roms/psx/`.

- **Button labels**
  - ○ Face buttons should use their graphical representation to avoid confusion, such as ⬡. Refer to [the Batocera-specific wiki syntax page](#) on how to use it.
  - ○ Other buttons should use path formatting (enclosed in '') so they aren't translated by the auto-translator, be capitalized as they appear on their printed label and be enclosed in square brackets within the path formatting to differentiate it from actual paths. For example, `[START]`, which is coded as ''[START]''.
  - ○ Shoulder buttons use their PlayStation format, such as `[R1]` and `[L2]`. Refer to the first-tier shoulder buttons as buttons, and the second-tier triggers as triggers.
  - ○ Due to ambiguity with `[L3]` and `[R3]`, mention "pushing in" when referring to pushing in the sticks. Eg. Then push in `[L3]` to switch mem-pak.
  - ○ If needing to note buttons where formatting is not available (such as in a code block), refer to the buttons by their cardinal direction followed by their equivalent SNES button in parenthesis. Eg. Press the South button (B SNES), then the START button.

- **Strings and menu items**
  - ○ Strings that need to be replaced by the user should be enclosed in less-than and greater-than symbols, followed by a single-sentence explanation of what's being replaced. Eg. Name the file `es_systems_<system name>.xml` where `<system name>` is the name of the system you want to customize.
  - ○ Menu items should be **CAPITALIZED AS THEY APPEAR IN ES AND BOLDED**, and menu navigation directions should have menu items separated by the arrow icon (→, coded as ->) enclosed in spaces. Eg. Go to **MAIN MENU** → **SYSTEM SETTINGS** → **AUDIO**. Title Case is also acceptable. Just be consistent with it.
  - ○ Items selected from within menus should be enclosed in double-quotes ("). Eg. From **Audio**, select "HDMI1".

- **Images**
  - ○ Should you embed an image that you are not the owner of, put credits to the original work/licensing information in "less significant" wrap <wrap lo> (can be applied with the Wrap plugin button in the toolbar). For example, Image illustrated by Wikimedia user [LucasVB](#). Licensed under [Creative Commons.](#)

Tense, tone and formality

This is more a subjective thing as everyone interprets tone and formality different. With that said, here are some loose guidelines:

- Use past tense for most things. Use current tense when giving instructions. For example: "has happened", "act on the thing", "navigate to", etc. Avoid future tense (unless explicitly referring to something that will be changed in the future). For example: don't use "is going to", "is happening", "acting on the thing", etc.
- Be objective and concise. Readers don't want to spend an inordinate amount of time shifting

through unnecessary information.
- Where possible, try to avoid abbreviations. Common, universal abbreviations are okay (for example, eg., etc., OS or URL). Abbreviations are acceptable if the full name has already been mentioned in the article.
- Not everyone will get/understand colloquialisms or sayings, avoid them when explaining critical information. They may also confuse the auto-translator. But with that said, a little humor is okay, especially if it's in a non-critical section/after the important information has been conveyed.
- Avoid referring to the reader. If the word "you" is used then this is referring to the reader; try rephrasing the sentence to be less addressing. It is understandable if information becomes too hard to convey following this guideline (there's an example above, about embedded image crediting).
- Avoid unnecessary manners such as "please" and "thank you". Although it might seem rude, it usually just adds too much fluff.

New/renaming pages

Renaming pages is highly discouraged. It should only be done in extreme cases where the original name fails to convey the content of the article. This is because search engines and links from external websites will break if the page name is changed. If absolutely sure that the rename is necessary, the page can be renamed with the "Rename page" button on the right-side toolbar (this will allow the wiki to update all the internal links to the new page automatically). A new page with the old page's name containing a link to the renamed page should be made to redirect search engines/old links.

Managing the sidebar

- The sidebar appears on all pages so its length should always be taken into consideration. Up until recently, the sidebar was always fully expanded showing all pages, but now it is collapsed by default easing the requirements for this.
- The sidebar currently has only one level of depth to it. More levels may be considered in the future, but for the moment stick with just using one level of depth. If you've added a very complicated topic with many pages to it, you may consider adding a whole new category for just it.
- Pages linked to in the sidebar should use internal links with the page's proper name as the display text. The page's proper name should use regular capitalization rules, not Camel Case like the page title is on the page itself.
- If you've renamed a page or outright removed it, check the sidebar to see if any links have broken. If so, either link them to the new name of the page or if you have removed the page, remove the link.
- Moving existing pages to other categories should be avoided, for consistent navigation, but if one category has become overloaded with too many pages it may be better to split that into multiple categories.

Categories are split by a collapsible section and individual pages are in an unordered list. Here's a short example of a category:

```
--> Getting Started#
  * [[:install_batocera|Install Batocera]]
```

```
   * [[:bios_files|BIOS files and how to add them]]
   * [[:add_games|Game ROMs and how to add them]]

<--
```

The sidebar syntax can be found on its section in the Batocera Wiki code snippets page.

Adding your page to the sidebar

If you've created a new page, the next job is to add a link to it to the sidebar page. You edit this page just like you would any other page.

- When considering which category to put it under, avoid adding anything to **Getting Started**. Put it in the category you believe would be the one a new user would use to find the information on your page. If adding a new system page in particular, it will be automatically linked in the systems overview page (as long as you've used the correct shortname) and doesn't need to be manually added.
- Pages linked to in the sidebar should use internal links with the page's proper name as the display text. The page's proper name should use regular capitalization rules, not Camel Case like the page title is on the page itself.
- New pages should also be added to the appropriate section in the navigation page along with a laconic description (if applicable). This is to assist those new to the wiki navigating to the appropriate page.

Individual elements

Headers

Headers are used to represent the basic structure of your page, you should always try to keep a clean structure in a page. The syntax for which can be found on the syntax page.

- The page title should the level 1 header, which uses six equal signs. There should only be one of these per page.
- Main sections should be split between level 2 headers, which use five equal signs. These should roughly be the same size between section, obviously there will be exceptions. If you're finding that the information is very disproportionately spread, consider sectioning the information according to a different criteria or moving some information to a more relevant page and linking to it using an in-link.
- Sub-sections should be split between level 3 headers. The proportion of these and lower sub-sections don't matter as much. You really shouldn't need to go further than this, try not to overuse sub-sections.
- There should be a blank line before and after your header.
- No two headers should share the same name as it makes it more difficult to link to, unless they are small enough sections such that linking to the header one-tier higher would make the meaning obvious anyway (this page for example).
- When deciding whether to use a header or a sub-header for a particular section of content, consider whether the content in the main header is **required knowledge** to understand the content being added. If it is, a sub-header is appropriate. If it is not, use a new main header.

Here is an example of a page's valid header code:

```
====== Playground ======
This page can be used as way to learn how to edit the wiki. We will first
explain why there is a dedicated page for it, and then why its URL has
playground:playground in the name
===== Goal =====
==== Training ====
As you can see, editing a wiki is not an easy task if you don't know how to
do it, so this page can be used for this purpose.
==== Avoid unnecessary rollbacks ====
While this wiki system can revert changes done to a previous version, asking
newcomers to edit actual pages would be risky, especially since some of them
aren't that easy to read and manage in the first place.
===== About the page name =====
The page name is playground:playground because the wiki is set up in such a
way that you can link pages to what is called a specific namespace, all the
emulator pages are under the "systems" namespace, so they all start with the
"systems:" name.
```

Table of contents

This wiki will automatically create a table of contents based of the headers on your page, unless you put ~~NOTOC~~ on top of your page. The ToC will dynamically change visible sub-sections as the user scrolls down the page. Unless your page has less than four headers total, you should always include a ToC.

Links

- Links to other websites should be linked with human-readable text showing where they lead to. For example, the Batocera home page!
- Links that refer to other wikis should use an inter-wiki link instead.
- Links that refer to other pages within this wiki should use an in-link instead.
- Links that refer to other sections in the same page should use a single-page section link. Protip: you can grab this by right-clicking the link icon when hovering over the section header, just cut off the initial `https://wiki.batocera.org/` part of the URL!
- If you've linked to a site once in a section, and mention the site again later in that section, you aren't required to also link that one. Mentioning it far later in another section on the same page, to the point that the user may no longer have the original link on their screen, it is encouraged to link it again.

Embedded images

Embedding images can greatly enhance the readers' understanding of the page, however there is such a thing as too much. When embedding images, always make sure you use as small a thumbnail size as possible while still having the image's contents be legible. This assists with mobile viewers. You can specify a size for thumbnails by appending ?500 at the end of your image URL, where the

number is how many pixels wide you want it to be.

Although this wiki has a white background, some users may be using custom css that renders the background as a dark color or even black (OLED phones typically enforce this). Keep this in mind when using images with transparency, and always add a white background (unless the image is being used for the explicit purpose of providing a high quality reference image, such as the license page).

Large images should be aligned to the left or center. Small and infrequent images can use any alignment. Multiple small images should be aligned to the right and have enough text content between them so that they don't vertically overlap and push each other toward the center. If there isn't enough text content, use center alignment instead to force the text content to render below the image.

Lists

Unordered lists (also referred to as bullet lists) should be used to list elements that can be interpreted in any order. Ordered lists (a.k.a. numbered lists) should be used to relay instructions that should be performed in a certain order.

Displaying code

Sometimes you'll need to display code as is without it being formatted by the wiki's markup.

- Code blocks should be used for multi-line code, and in-line should be used for single strings of code.
- The <code> keyword should always be on its own line for clarity.

  ```
  With that said, if code is needed to be displayed in a bullet list, the
  first line must begin in the bullet point in the list item's line.
  Secondary lines and the closing brackets in the code don't have to also
  follow this rule, but the next bullet point should be the next
  immediate line.
  ```

- The only exception to <code> being in the middle of a line is if the list needs to continue afterward, as above. View this page's code for an example.

Path and commands

To write down paths, or commands, it is fine to surround it with two apostrophes (''), so ''/userdata/system/batocera.conf'' will display as /userdata/system/batocera.conf. Doing this will not skip formatting by itself however, so keep that in mind. You can skip formatting by putting in two percentage symbols like this:

```
''%%--command [option]%%''
```

Use this when specifying a single command within a sentence. If you need to specify multiple commands within a single sentence, consider using a code block instead. Eg. Then run batocera-save-overlay

Tables

- You should only use tables when comparing multiple multi-attribute items which share the same category for a particular attribute. For example, if you need to compare a bunch of different emulators for a system and compare their readable file types/internal name/additional notes about function.
- Tables should not be used for a single-attribute list of items, use unordered bullet lists instead.
- Tables can be handy for image comparison, just keep in mind the scaling won't necessarily reflect your intended resolution.

This wiki will automatically adjust the spacing and scaling of tables based on their content and the amount of screen-space available, so no need to worry about that yourself.

Collapsible sections

Should be avoided in most if not all cases. Collapsible sections require the user to interact with the page, which may not be compatible with certain devices, be difficult to employ, or seem amateurish. With that said, collapsible sections can be used to organize inordinate amounts of necessary information that can be irrelevant to the user currently scanning through the page, such as all the topics available in the sidebar or a really large configuration file.

If you're not quite sure

Compare it to how other similar pages have done it! The beauty of the wiki is that you will be covering information that hasn't been covered before, so there may not be a consistent formatting for it yet. Look at how other, similar-enough pages have formatted their information and take it from there.

For making a new system page specifically, you can follow the system template in the playground.

From:
https://wiki.batocera.org/ - **Batocera.linux - Wiki**

Permanent link:
https://wiki.batocera.org/wiki:wiki-editing-practices

Last update: **2023/03/02 23:47**