1 # Rocky Linux Documentation How-To Contribute Guide
2
3 ## Introduction
4
5 So you want to contribute to the Rocky Linux documentation, but aren't sure how to go about that? You've come to the right place. We're excited to have you on board with us.
6
7 One of our big goals for Rocky is to have not just good, but GREAT documentation, and we can't do it alone. This document will get you started, and will (hopefully) allay any worries you might have about doing so. We're always here to help, too, so feel free to ask questions and join in the conversation.
8
9 ## Documentation Subject Matter
10
11 If you can dream it âM−^@M−^T or better yet, have done it before âM−^@M−^T then your subject matter is probably important to the Rocky Linux documentation project. Once a build is finally out there, then we expect that documentation will really start to explode, but we want to be ready with some things already documented out of the gate.
12
13 This is where you come in âM−^@M−^T both now and in the future. If you have experience with anything Linux−related, and can translate that to Rocky Linux, then we want you to start writing!
14
15 ## Where to Start
16
17 While you can create documentation in any format, the preferred format is Markdown. You can easily create documents in Markdown, plus Markdown is super−easy to learn. The best way to get started with Markdown is to have a Markdown editor installed and read the tutorial. You can create Markdown files with any text editor, but most Markdown editors allow you to preview what you've input already, so they can be very helpful.
18
19 ### Markdown Editor
20
21 As we've said already, the best way to create Markdown files, particularly if you've not done it before, is to grab an editor for the operating system that you use on your PC or Laptop. A simple web search for "Markdown editors" will show you a bunch to choose from.
22
23 Pick one that is compatible with your OS and get started. This particular How−To doc was created with [ReText](https://github.com/retext-project/retext), a cross-platform Markdown editor. Again, if you'd prefer to create your Markdown files in your own text editor that you are already familiar with, that's no problem at all.
24
25 #### Alternate Markdown Editors
26
27 ReText is good, but if you feel like exploring your other options, there are a few:
28
29 * [Zettlr](https://www.zettlr.com) − Free, cross-platform, and open source
30 * [Mark Text](https://marktext.app) − Free, cross-platform, and open source
31 * [ghostwriter](https://wereturtle.github.io/ghostwriter/) − Free, Windows and Linux, open source
32 * [Remarkable](https://remarkableapp.github.io) − Linux-only, open source
33 * [Typora](https://typora.io) − Free and cross-platform, but *not* open source
34
35 ### Markdown Tutorial
36
37 You can get a good feel for how to write Markdown files, by taking a look at [Mastering Markdown](https://guides.github.com/features/mastering-markdown/). This online resource will get you up-to-speed in no time.
38
39 ## Using Git
40
41 Rocky Linux, like loads of other projects out there, uses "git" to manage its code and files, including documentation files. This tutorial assumes a very basic knowledge of git, and how it works. The second example also presupposes familiarity with the command line.
42
43 The steps listed below will get you started. In the meantime (and until we inevitably develop our own guide), you can learn Git in-depth with this [free Udacity course](https://www.udacity.com/course/version-control-with-git--ud123). If you don't have the time for that, here's [a shorter guide](https://blog.udacity.com/2015/06/a-beginners-git-github-tutorial.html).
44
45 In order to submit your text, we ask that you create a GitHub account. When you are ready to submit your written documentation for approval, follow these easy steps:
46
47 ### With the GitHub GUI
48
49 Most tasks can be completed from the web GUI on GitHub. Here's an example of adding a file you've created on your local machine to the Rocky Linux documentation GitHub repository.
50
51 1. Login to your GitHub account.
52 2. Navigate to the Rocky Linux Documentation repository at: https://github.com/rocky-linux/documentation.git
53 3. You should be on the **"Main"** branch, so check the drop down label down in the middle section, just to be sure you are. Your document might not ultimately end up in the **"Main"** branch, but admins will move it around to where it logically fits later.
54 On the right−hand side of the page, click the "Fork" button, which will create your own copy of the documentation.
55 4. In the middle of the page on the forked copy, just to the left of the Green "Code" drop down, is an "Add file" button. Click this and choose the "Upload files" option.
56 5. This will give you a way to either drag and drop files here, or to browse to them on your computer. Go ahead and use the method you are most comfortable with.
57 6. Once the file is uploaded, the next thing you need to do is create a Pull Request. This request lets the upstream administrators know that you've got a new file (or files) that you would like them to merge with the master.
58 Click on "Pull Request" in the upper−left of the screen.
59 7. Write a brief message in the "Write" section letting the administrators know what you've done. (New document, revision, suggested change,

```
     etc.,) then submit your change.
60
61   ### From the Git Command-Line
62
63   If you prefer to run Git locally on your own machine, you can clone the Rocky Linux Documentation repository, m
     ake changes, and then do a commit changes afterwards. To make things super-easy, do steps 1-3 on *From Git Gui*
      above then:
64
65   1. Git clone the repository: `git clone https://github.com/your_fork_name/documentation.git`
66   2. Now, on your own machine, add files to the directory.
67   Example: `mv /home/myname/help.md /home/myname/documentation/`
68   3. Next, run git add for that file name.
69   Example:  `git add help.md`
70   4. Now, run git commit for the changes you have made.
71   Example: `git commit -m "Added the help.md file"
72   5. Next, push your changes to your forked repository: `git push https://github.com/your_fork_name/documentation
      main`
73   6. Next, we repeat step 6 and 7 above: ...create a Pull Request. This request lets the upstream administrators
     know that you've got a new file (or files) that you would like them to merge with the master branch. Click on "Pull Request" in the upper-left of the screen
74
75   Once all of that is done, you simply wait for confirmation that your request has been successfully merged. (Or not, as the case may be.)
76
77   ## Keep Up With The Conversation
78
79   If you haven't already, join the conversation on the [Rocky Linux Mattermost Channel](https://chat.rockylinux.org/
     rocky-linux/) and stay up-to-date with what is going on. Join the "Documentation" channel, or any other channel y
     ou are interested in. We'd be glad to have you with us!
80
```