

Redmine - Defect #22323

Markdown newline rendering broken

2016-03-26 17:13 - Viktor Berke

Status:	New	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Text formatting	Estimated time:	0.00 hour
Target version:	Candidate for next major release	Affected version:	3.2.1
Resolution:			
Description			
<p>As per the specs Markdown should not insert <code>
</code> for single line breaks. <code>
</code> should only occur for:</p> <ul style="list-style-type: none">- A line ending with 2 or more spaces, then a line break.- One or more blank lines between paragraphs lines. In other words, 2 or more line breaks. <p>GitHub is consistent with this in wiki formatting. In GitHub issues a single line break also triggers a rendered line break, the reason for this is beyond me but that's also against the specs.</p> <p>Please fix this. This actually breaks Pandoc conversion from Textile pages because Pandoc occasionally inserts line breaks even within paragraphs and this causes line breaks in the rendered text as well thus breaking the format.</p>			
Related issues:			
Related to Redmine - Feature # 22005: Rake task for converting from Textile t...		New	
Related to Redmine - Patch # 32424: CommonMark Markdown Text Formatting		New	

History

#1 - 2016-03-26 17:18 - Viktor Berke

Since I can't add related issues, I'll just link it: [Rake task for converting from Textile to Markdown](#)

#2 - 2016-03-27 04:01 - Go MAEDA

- Category changed from Wiki to Text formatting

#3 - 2016-04-25 06:11 - Toshi MARUYAMA

- Related to Feature #22005: Rake task for converting from Textile to Markdown added

#4 - 2019-05-10 12:36 - Pablo Yanez Trujillo

I'd like to know whether this has been fixed in the current 4.x releases. I still have an 3.3.7 installation and this is annoying, specially since I use text-aid-too (<https://github.com/smblott-github/text-aid-too>) to edit content with vim and I have line wrapping at 80 characters. When editing on github, or stackoverflow then there is no problem. But on my redmine pages, it's a problem, because redmine inserts `
` where it shouldn't.

#5 - 2019-08-20 16:29 - Martin Cizek

- File 22323-with-test.patch added

On one hand, calling this a defect might not be fair, as it is apparently kind-of decision made in `lib/redmine/wiki_formatting/markdown/formatter.rb` with the setting `:hard_wrap => true`. Indeed, we do not know how intentional the decision was.

On the other hand, it breaks compatibility with GFM, especially on its two major implementations GitHub and GitLab (as said - except GitHub issues, but our developers are really angry about GitHub issues having different behaviour than the rest of GitHub). And as the author says, it is against Markdown specs.

I guess we are not that rare when we intend to share Markdown sources between Redmine and GitLab and among Redmine and other systems - using API, mirroring wiki to git and indeed with users' copy&pasting it.

So I would strongly vote to change the default behavior towards more standard Markdown / GFM to stay compatible with the world.

The patch is included including a unit test. :) P.S. The tests at the end of MarkdownFormatterTest seem to be commented out by private? Not sure why, but the tests work even after commenting it out to #private.

#6 - 2019-08-20 19:39 - Mischa The Evil

Martin Cizek wrote:

P.S. The tests at the end of MarkdownFormatterTest seem to be commented out by private? Not sure why, but the tests work even after commenting it out to #private.

I have reported this in a separate issue, #31929. Thanks for mentioning it here.

#7 - 2019-08-22 15:03 - Martin Cizek

- File 22323-with-test-after-r18381.patch added

Mischa The Evil wrote:

I have reported this in a separate issue, #31929. Thanks for mentioning it here.

You're very welcome. Attaching the patch updated against trunk (r18381).

#8 - 2019-09-02 12:48 - Go MAEDA

- Target version set to Candidate for next major release

#9 - 2019-12-08 14:48 - Martin Cizek

Target version set to Candidate for next major release

Now, when there is #32424, I'd vote for keeping the markdown format as is and deprecate it. This would be actually a cleaner solution for users who rely on its current behavior, which includes hard line breaks.

#10 - 2020-05-22 09:02 - Mischa The Evil

- Related to Patch #32424: CommonMark Markdown Text Formatting added

Files

22323-with-test.patch	1.47 KB	2019-08-20	Martin Cizek
-----------------------	---------	------------	--------------

