Redmine - Feature #20497

Markdown formatting supporting HTML

2015-08-08 09:47 - dumb blob

Status:	Closed		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	0%
Category:	Text formatting		**Estimated time:**	0.00 hour
Target version:				
Resolution:	Wont fix			

Description

As discussed in [#15520#note-16](#) currently HTML is disabled. This brings several **important issues** which need to be addressed:

- Markdown without HTML is not any more Markdown, but rather a very limited flavour
- because it's a flavour, it needs to be named differently (according to *John Gruber*)
- users of CKEditor (or similar) can't easily migrate
- Redmine is very often deployed on non-public networks and thus using HTML is quite safe

The easiest solution would be to introduce 2 formatting options instead of one:

1. "Markdown (with HTML)"
1. "Markdown (without HTML)"

Also, there would need to be two different markup manuals (a modern variant based on side-by-side live preview as on http://agea.github.io/tutorial.md/ might be worth looking at and possibly copying from - it's MIT licensed).

Related issues:

Related to Redmine - Defect #807: HTML not escaped in ticket descriptions	**Closed**	**2008-03-07**
Related to Redmine - Feature #15520: Markdown formatting	**Closed**	
Related to Redmine - Feature #32424: CommonMark Markdown Text Formatting	**Closed**	

History

#1 - 2015-08-08 10:22 - Go MAEDA

- Related to Defect #807: HTML not escaped in ticket descriptions added

#2 - 2015-09-06 09:08 - Toshi MARUYAMA

- Description updated

#3 - 2015-10-07 20:55 - @ go2null

Useful background discussion here [#15520](#) *Markdown formatting*, including links to the following source for [Textile](#) and [Markdown](#).

#4 - 2015-10-08 01:27 - Go MAEDA

- Related to Feature #15520: Markdown formatting added

#5 - 2015-10-08 01:57 - Go MAEDA

HTML tags except <pre> were disabled by [#807](#).

#6 - 2015-10-08 01:57 - Go MAEDA

- Priority changed from High to Normal

#7 - 2015-10-08 18:45 - @ go2null

Here is StackOverflow's disclaimer:
http://stackoverflow.com/editing-help#html

===============
Inline HTML

If you need to do something that Markdown can't handle, use HTML. Note that [we only support a very strict subset of HTML!](#)

```
To reboot your computer, press ctrl<u>alt</u>del.
```

Markdown is smart enough not to mangle your span-level HTML:

```
<b>Markdown works *fine* in here.</b>
```

Block-level HTML elements have a few restrictions:

1. They must be separated from surrounding text by blank lines.
2. The begin and end tags of the outermost block element must not be indented.
3. Markdown can't be used within HTML blocks.

```
<pre>
    You can <em>not</em> use Markdown in here.
</pre>
```

===============

And here are the allowed HTML tags:

```
<a>              - hyperlink.
<b>              - bold, use as last resort <h1>-<h3>, <em>, and <strong> are
                   preferred.
<blockquote>     - specifies a section that is quoted from another source.
<.code>          - defines a piece of computer code.
<del>            - delete, used to indicate modifications.
<dd>             - describes the item in a <dl> description list.
<dl>             - description list.
<dt>             - title of an item in a <dl> description list.
<em>             - emphasized.
<h1>, <h2>, <h3> - headings.
<i>              - italic.
<img>            - specifies an image tag.
<.kbd>           - represents user input (usually keyboard input).
<li>             - list item in an ordered list <ol> or an unordered list <ul>.
<ol>             - ordered list.
<p>              - paragraph.
<.pre>           - pre-element displayed in a fixed width font and and
                   unchanged line breaks.
<s>              - strikethrough.
<sup>            - superscript text appears 1/2 character above the baseline
                   used for footnotes and other formatting.
<sub>            - subscript appears 1/2 character below the baseline.
<strong>         - defines important text.
<strike>         - strikethrough is deprecated, use <del> instead.
<ul>             - unordered list.
<br>             - line break.
<hr>             - defines a thematic change in the content, usually via a
                   horizontal line.
```

#8 - 2015-11-23 19:51 - James H

i run our redmine on a non-public network and would like to use this feature, but for our use case, we would need to be able to specify which projects would have this enabled or disabled. Most of our users are not "power" users (do not know html) and so only a limited number of projects would get this setting enabled.

It would also be great if it could be even more customize-able than that (by trackers, by users, etc.).

#9 - 2016-07-08 11:31 - Anonymous

+1

#10 - 2016-07-14 15:14 - Ben Blanco

I concur that **Markdown without HTML** is a flavoured(down) version of Markdown.

Github.com also rely on redcarpet gem, but they first sanitize the raw input before passing it onto redcarpet, see https://github.com/github/markup#github-markup (code is under Github's *"BSD-like?"* license).

The sanitization is done by html-pipeline which is available as a gem, (its code is under MIT license).

By doing so they allow fair set of HTML tags:

```
WHITELIST = {
  :elements => %w(
    h1 h2 h3 h4 h5 h6 h7 h8 br b i strong em a pre code img tt
    div ins del sup sub p ol ul table thead tbody tfoot blockquote
    dl dt dd kbd q samp var hr ruby rt rp li tr td th s strike summary details
  ),
```

Note: code excerpt from html-pipeline's [sanitization_filter.rb](#) *at line 44.*

Does anyone on this thread, or devs at redmine.org, think that it could be an option to implement the same process for redmine? And/or even re-use Github's code?

Last, [#20497#note-8](#) made me think - though maybe not a good idea (that's why I ask) - if redmine's permissions model could be used to handle who has the right to input HTML tags in Wiki/Issue markdown.. Not sure if it would be better/more flexible than allowing HTML input on a per-project-basis...or have both options?

In all cases, being able to use HTML in redmine w/ markdown (in my case first and foremost for building better <tables> than Markdown's syntax allows for) would be awesome!

Thx :)

#11 - 2016-07-14 21:07 - Anonymous

Ben Blanco wrote:

> In all cases, being able to use HTML in redmine w/ markdown (in my case first and foremost for building better <tables> markdown's syntax allows for) would be awesome!
>
> Thx :)

That is exactly the same reason I like to allow markup in my Markdown input.. I disabled the filter_html rule in the [core formatting rules](#) for my privately hosted Redmine instance to allow that, so that's sorted it out for me. I like the idea of making this behaviour configurable, I believe I may have suggested the same thing in some of the other Markdown-related tickets too.

Just keep in mind, as per [mc0e's reply](#) to my question on this matter, if you're running a public facing Redmine server you will have some security concerns to consider. For LAN or WAN only servers (like in my case) this isn't really a problem.

#12 - 2016-07-18 10:09 - Adrien Crivelli

HTML in markdown is also required to solve the cases of two consecutive lists, or a list followed by code block as [described in details in pandoc manual](#). Basically we use HTML comment (<!-- -->) to mark the end of a list when necessary. This is also [explained here](#), where it's stated that [all Markdown libs behave the same](#) and that there is no other solution than using HTML comment.

So at the very least a subset of HTML really seems to be a necessity. Tables and comments being the obvious one.

#13 - 2019-03-15 20:04 - Dan Hi

This has been mentioned in others issues pointing at this one, but my [#1](#) reason for wanting HTML would currently be the resizing of images or using thumbnails. Coming from JIRA where thumbnails are quite easy.

See here

https://stackoverflow.com/questions/14675913/changing-image-size-in-markdown

"With certain Markdown implementations (including Mou and Marked 2 (only macOS)) you can append =WIDTHxHEIGHT after the URL of the graphic file to resize the image. Do not forget the space before the =."

The lack of this sent me towards the html option, which honestly is part of true markdown. I'm not sure why it's so hard to rename markdown in redmine to "redmine markdown"--that's a bare minimum of what should be done here. I'm not interested in hacking the rbs to allow HTML, only to lose it after an update.

If not HTML, then image resizing. And in either case, call it "redmine markdown" to avoid confusion about what it supports.

#14 - 2019-03-16 09:32 - Bernhard Rohloff

Some time ago I've made a plugin to add a macro for that purpose as proposed in [RE: how to allow html tag in redmine wiki](#). The syntax is quite easy and there's no need for an extra setting. I can attach it here on Monday if there's demand for it.

#15 - 2019-10-23 09:20 - Jens Krämer

- File 0001-adds-an-additional-Markdown-format-that-allows-user-.patch added

Here's a patch that adds a second *Markdown (with HTML)* formatter which allows HTML input like tables etc and sanitizes the resulting HTML using Rails' HTML sanitization tools to prevent XSS etc. I have no idea how well that would work for handling CKEditor content, ymmv.

That's more or less just a proof of concept - I dont think that having two Markdown flavors is really a good idea. Ultimately, I don't see a problem with generally allowing HTML for the standard *Redmine Markdown* formatting as long as the output is sanitized properly. One might think about using the sanitize gem like Github does instead of Rails' Loofah based sanitizer as it allows for more granular definition of rules and is potentially safer / better due to the different HTML/CSS parsers used. Also, Rails' sanitize strips HTML comments which makes this patch useless if you want to support the HTML comment based list separation technique mentioned above :) However adding <p></p> between two lists serves the same purpose (and even works with Redmine's stock *Markdown* formatter).

I like the html-pipeline approach a lot, but introducing that to Redmine core would imo only make sense as part of a bigger refactoring / modularization of the whole textile/markdown rendering.

#16 - 2019-10-30 10:38 - Go MAEDA

- Target version set to Candidate for next major release

#17 - 2019-11-06 11:12 - Jan from Planio www.plan.io

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

#18 - 2021-08-12 23:02 - Marius BĂLTEANU

- Status changed from New to Closed

- Target version deleted (Candidate for next major release)

- Resolution set to Wont fix

The new CommonMark Markdown (Github Flavoured) that will be available in 5.0.0 supports HTML. I'm closing this as "Won't fix" because the current Markdown implementation based on RedCarpet is going to be dropped in the future versions.

Migrating to the new engine it's recommended. Feel free to open new issues for the missing features.

Files

0001-adds-an-additional-Markdown-format-that-allows-user-.patch	14.4 KB	2019-10-23	Jens Krämer