



Redmine - Feature #27758

Adds preview option to the wiki toolbar

2017-12-07 21:41 - Marius BĂLTEANU

Status:	Closed	Start date:	
Priority:	Normal	Due date:	
Assignee:	Jean-Philippe Lang	% Done:	0%
Category:	UI	Estimated time:	0.00 hour
Target version:	4.0.0		
Resolution:	Fixed		
Description			
<p>I'm working on a patch that adds the Preview functionality to the wiki toolbar. I find it more friendly to have the preview option for each individual field, instead of the bottom link. Most of the time, when I'm writing the description of an issue, I need to scroll down to the preview link, click it, view the preview and then scroll up to continue writing.</p> <p>I want to share some screenshots in order to take some feedback from the community.</p> <p>1. Write/Preview tabs:</p> <p> When the preview tab is opened, the textarea field is replaced with the formatted text. In this way, you can easily switch between write/preview modes.</p> <p>2. Preview button</p> <p> Instead of the tabs, we can have just a button that toggles between Preview and Write.</p>			
Related issues:			
Related to Redmine - Feature #27368: Make preview for comments for news		Closed	
Related to Redmine - Feature #2604: Preview comment on news / bulk edit		Closed	2009-01-28
Related to Redmine - Feature #22015: A side by side textile preview would by ...		Closed	
Related to Redmine - Feature #9174: Suggestion - Preview on Documents		Closed	2011-09-02
Related to Redmine - Feature #4252: Move preview button to before "submit"		Closed	2009-11-20
Related to Redmine - Defect #29694: Some buttons in wiki toolbar are hidden i...		New	
Related to Redmine - Feature #29712: Preview and wiki toolbar for full width ...		Closed	
Related to Redmine - Defect #30988: Preformatted text overflows the preview area		Closed	
Related to Redmine - Defect #30758: Preview URL in Wiki Toolbar should be esc...		Closed	
Related to Redmine - Feature #30459: Switch edit/preview tabs with keyboard s...		Closed	
Related to Redmine - Patch #34444: Remove unused key :preview from Redmine::A...		Closed	
Has duplicate Redmine - Feature #7157: Content Editor Preview at the same pla...		Closed	2010-12-22

Associated revisions

Revision 17521 - 2018-09-26 09:27 - Jean-Philippe Lang

Adds preview option to the wiki toolbar (#27758).

Patch by Marius BALTEANU.

Revision 17524 - 2018-09-26 11:13 - Jean-Philippe Lang

Removed unused image (#27758).

Revision 17525 - 2018-09-26 12:45 - Jean-Philippe Lang

Use Edit instead of Write (#27758).

Patch by Marius BALTEANU.

Revision 17526 - 2018-09-26 12:58 - Jean-Philippe Lang

Adds strings to jstoolbar locales (#27758).

Revision 17527 - 2018-09-26 13:00 - Jean-Philippe Lang

Removed unused helper #preview_link (#27758).

Revision 17530 - 2018-09-28 00:08 - Go MAEDA

Add label_nothing_to_preview to locales (#27758).

Revision 17541 - 2018-09-30 11:45 - Jean-Philippe Lang

Adds style to the empty preview message (#27758).

Revision 17542 - 2018-09-30 11:47 - Jean-Philippe Lang

Moves style to jstoolbar.css (#27758).

Revision 17546 - 2018-09-30 18:12 - Jean-Philippe Lang

Removes space between tabs and textarea (#27758).

Revision 17548 - 2018-09-30 18:33 - Jean-Philippe Lang

Fixed that changing tabs in project settings unselect "Write" tab of project description textarea (#27758).

History

#1 - 2017-12-07 21:42 - Marius BĂLTEANU

- *Subject changed from Adds preview tab to the wiki toolbar to Adds preview option to the wiki toolbar*

#2 - 2017-12-08 00:20 - Holger Just

I really like this proposal as it makes it much more apparent when a preview is available. Using a tab directly besides the respective text field helps users to not lose focus of their work.

UI-wise, I like the tab-approach (your first screen shot) much better than the button since the tabs make it clear that you change between different views of the same thing (the entered text). This idiom is also common in other software like GitHub or Wordpress. I'd like to propose some slight changes though:

- We should use the exact visual design of the existing tabs elements already used elsewhere, e.g. in the revision view of the repository where it's possible to switch between the list of changes and the diff or the project's settings tab.
- I think those edit icons above the text box should be part of the the "Write" tab itself so that they are removed when you switch to the "Preview" tab. Here, we might need to be a bit creative since it is probably desirable to not introduce another "row" of text (tab, icons, text box). It might be possible to combine the icons and the tabs somehow while still sticking to the existing components used elsewhere in Redmine.

Generally, the layout should also work when there is no explicit preview option for the text box available since this might not strictly be the case for all fields; the welcome text in the settings comes to mind, as well as e.g. text custom fields or stuff added by plugins. It would however be awesome if this preview / tab bar would be as versatile as the existing icon bar, preferably it could even be added with exactly the same code (or just small changes).

All in all though, I like this very much!

#3 - 2017-12-08 18:43 - Marius BĂLTEANU

Holger Just wrote:

I really like this proposal as it makes it much more apparent when a preview is available. Using a tab directly besides the respective text field helps users to not lose focus of their work.

Thanks for your feedback, Holger.

UI-wise, I like the tab-approach (your first screen shot) much better than the button since the tabs make it clear that you change between different views of the same thing (the entered text). This idiom is also common in other software like GitHub or Wordpress. I'd like to propose some slight changes though:

- We should use the exact visual design of the existing tabs elements already used elsewhere, e.g. in the revision view of the repository where it's possible to switch between the list of changes and the diff or the project's settings tab.

Totally agree.

- I think those edit icons above the text box should be part of the the "Write" tab itself so that they are removed when you switch to the "Preview" tab.

Yes, I've already implemented this feature.

Here, we might need to be a bit creative since it is probably desirable to not introduce another "row" of text (tab, icons, text box). It might be possible to combine the icons and the tabs somehow while still sticking to the existing components used elsewhere in Redmine.

For now, I chose to create the tabs from javascript, in the same way how the toolbar buttons are created and most of the changes are made in the `jstoolbar.js` file. If you think that is better to create the tabs using the existing `render_tabs` method, I think that I can find a solution.

Generally, the layout should also work when there is no explicit preview option for the text box available since this might not strictly be the case for all fields; the welcome text in the settings comes to mind, as well as e.g. text custom fields or stuff added by plugins. It would however be awesome if this preview / tab bar would be as versatile as the existing icon bar, preferably it could even be added with exactly the same code (or just small changes).

In the current implementation, the Preview works wherever the `wikitoobar_for` method is called because I've added a default method `preview#text`.

All in all though, I like this very much!

I'm going to post this weekend a first version of my patch and I think that we can discuss more about the technical solution after that.

#4 - 2017-12-11 19:25 - Marius BĂLTEANU

- *File write.png added*

- *File preview.png added*

- *File write_preview_tabs_wip.patch added*

Holger, attached is a working version (only for markdown) of my patch. I still have some things to do (like adding support for textile, removing preview links, add tests), but if you're interested to test it, is enough. The functionality and the design are the final one (attached also 2 screenshots). I'm not 100% happy with the results from UI perspective, but is the best that I can do with my current CSS skills, using the existing Redmine components and not adding too many changes.

Any feedback is really appreciated. My intention is to have a final version of this patch until 21.12.

Write tab:

`write.png`

Preview tab:

`preview.png`

#5 - 2017-12-12 02:56 - Go MAEDA

- *File 27758-many-lines.png added*

Thank you for working on this feature. I think that this is an awesome improvement.

But I noticed that the preview is not perfectly displayed when I enter text with many lines. Maybe we need a vertical scroll bar or a responsive preview area.

Notes

Write Preview

01: Quick brown fox jumps over the lazy dog.
 02: Quick brown fox jumps over the lazy dog.
 03: Quick brown fox jumps over the lazy dog.
 04: Quick brown fox jumps over the lazy dog.
 05: Quick brown fox jumps over the lazy dog.
 06: Quick brown fox jumps over the lazy dog.
 07: Quick brown fox jumps over the lazy dog.
 08: Quick brown fox jumps over the lazy dog.
 09: Quick brown fox jumps over the lazy dog.
 10: Quick brown fox jumps over the lazy dog.
 11: Quick brown fox jumps over the lazy dog.
 12: Quick brown fox jumps over the lazy dog.
 13: Quick brown fox jumps over the lazy dog.
 14: Quick brown fox jumps over the lazy dog.
 15: Quick brown fox jumps over the lazy dog.
 16: Quick brown fox jumps over the lazy dog.
 17: Quick brown fox jumps over the lazy dog.
 18: Quick brown fox jumps over the lazy dog.

Submit Preview Cancel

Also available in: [Atom](#) | [PC](#)

Powered by [Redmine](#) © 2006-2017 Jean-Philippe Lang

#6 - 2017-12-12 08:48 - Marius BĂLTEANU

Go MAEDA wrote:

But I noticed that the preview is not perfectly displayed when I enter text with many lines. Maybe we need a vertical scroll bar or a responsive preview area.

Thanks. I've fixed the issue on my local development.

#7 - 2018-01-07 10:10 - Mischa The Evil

I'll start by saying that I'm also in favor of an overall change as here proposed. That said, I just came around another UI-wise slightly similar plugin implementation (mostly done using JavaScript), where seeing its screenshots reminded me about this issue: https://github.com/leish/redmine_editor_preview_tab. The respective screenshots:

- Write tab: `screenshot_write.png`
- Preview tab: `screenshot_preview.png`

Comparing the plugin with your implementation as of note#4 (further referred to as '[the] patch':

- I think that a UI-design like provided by your patch is preferred, for the same reasons as mentioned by Holger in note#2;
- your patch is already more complete/polished (think e.g. access-keys);
- your patch is a bit less JavaScript heavy;
- your approach of implementing this in a similar fashion as (other) toolbar buttons is interesting, creative and (visually) neat.

Nice job. I'm interested in seeing subsequent iterations of this patch.

#8 - 2018-01-08 20:07 - Marius BĂLTEANU

- File `drop-jstoolbar-textile.min.patch` added

- File `replace_preview_link_with_a_write_preview_tabs.patch` added

I've finished most of the work for this feature. UI tests are missing currently, but I'll add a new patch with them next days. Until then, the patches can be tested and/or reviewed.

1. `drop-jstoolbar-textile.min.patch`:

This patch removes the `jstoolbar-textile.min.js` file and uses for textile formatting the same `jstoolbar.js` file. I chose to remove this file because I wasn't able to figure out which is the "official" way to generate a new merged and minified version from the new `jstoolbar`. In the same time, if it is desired to use a minified version (which is a good idea), we should document the tools and the steps to generate it and we should apply the same logic also for markdown formatting (at least).

2. `replace_preview_link_with_a_write_preview_tabs.patch`:

- adds the write preview tabs to the text formatting toolbar

- removes all the existing Preview links
- adds a default preview url that can be used to preview simple texts (eg: project description from the project settings)
- on responsive mode, when the screen resolution width is less than the total width of the js toolbar (tabs + buttons), some buttons become hidden. I'm not happy with the current solution, but is not a blocker from my point of view and we can improve this later when we have a better solution.

Considering that the changes are quite big, I think that it is a good opportunity to discuss the implementation of this feature in Redmine [4.0.0](#).

#9 - 2018-01-17 10:35 - Marius BĂLTEANU

- Related to Feature #27368: Make preview for comments for news added

#10 - 2018-01-17 10:36 - Marius BĂLTEANU

- Related to Feature #2604: Preview comment on news / bulk edit added

#11 - 2018-01-17 10:47 - Marius BĂLTEANU

- Related to Feature #22015: A side by side textile preview would be very helpful added

#12 - 2018-01-17 10:49 - Marius BĂLTEANU

- Related to Feature #9174: Suggestion - Preview on Documents added

#13 - 2018-01-17 10:50 - Marius BĂLTEANU

- Related to Feature #7157: Content Editor Preview at the same place where text is added

#14 - 2018-01-17 10:52 - Marius BĂLTEANU

- Related to Feature #4252: Move preview button to before "submit" added

#15 - 2018-01-26 20:46 - Marius BĂLTEANU

- File deleted (replace_preview_link_with_a_write_preview_tabs.patch)

#16 - 2018-01-26 20:47 - Marius BĂLTEANU

- File deleted (write_preview_tabs_wip.patch)

#17 - 2018-01-26 21:13 - Marius BĂLTEANU

- File write.png added

- File preview.png added

Attached is a new version of the patch which includes:

- fixes for the existing tests
- new system tests for the write/preview feature in various screens
- small fixes

This version of the patch is final from my point of view.

Some screenshots:

Write tab:

write.png

Preview tab:

preview.png

#18 - 2018-01-26 21:14 - Marius BĂLTEANU

- File replace_preview_link_with_write_preview_tabs.patch added

#19 - 2018-01-29 18:15 - Bernhard Rohloff

I've played around with your latest patches and I haven't found any flaw, or unexpected behavior. Except for the missing accesskey 'r' to switch to the preview.

It's definitely better than the mentioned plugin which I have installed on my private Redmine and a great improvement to Redmine's UX.

I'm looking forward to see this merged, soon.

#20 - 2018-01-30 04:49 - Mischa The Evil

- Assignee set to Marius BĂLTEANU

- Target version set to Unplanned backlogs

Marius BALTEANU wrote:

1. drop-jstoolbar-textile.min.patch:

This patch removes the jstoolbar-textile.min.js file and uses for textile formatting the same jstoolbar.js file. I chose to remove this file because I wasn't able to figure out which is the "official" way to generate a new merged and minified version from the new jstoolbar. In the same time, if it is desired to use a minified version (which is a good idea), we should document the tools and the steps to generate it and we should apply the same logic also for markdown formatting (at least).

I agree completely on this. See eg. [#14937](#) where this became troublesome too.

Bernhard Rohloff wrote:

[...] Except for the missing accesskey 'r' to switch to the preview.

This seemed to be working in the patch I reviewed ~~looked at~~ as posted in note#4.
[Marius Ionescu](#): can you take a look at this?

#21 - 2018-01-31 20:56 - Marius BĂLTEANU

[Bernhard Ganslmeier](#), thanks for testing these patches.

Regarding the missing accesskey 'r', I'm aware, but I can't make it work correctly (using only the native accesskey attribute) when we have more than one JS Toolbar in the page (like in the issue#edit mode). Hitting the 'r' key when the focus is in the textarea, will switch the preview tab from the next JS Toolbar because is the next element in the page that has the respective accesskey. To make it more clear, look at the HTML structure (illustrated below) from the issue#edit and observe that the textarea is always after its own Preview element.

- ...
- write tab
- preview tab with accesskey 'r'
- textarea for description field
- ...
- write tab
- preview tab with accesskey 'r'
- textarea for notes field

To make it work as expected, I need to override the native functionality using javascript, but this is something bigger and I want to do it in the future because:

- I would like to add some shortcuts for accessing the jstoolbar buttons (Control/Command + B for bold, Control/Command + I for italic, etc..).
- I will propose to use a JS library that implements a consistent shortcut key between the browsers. I find it very difficult to use the current implementation that depends on the browser and OS.

From my point of view, this missing accesskey should not be a blocker for this ticket.

#22 - 2018-02-01 06:25 - Bernhard Rohloff

I think the missing accesskey is no big deal, because this patch lets you access the previews easier than the mostly unknown accesskey did. I also injected the accesskey in the tab with the development tools of Firefox to get a feeling for the UX and it wasn't good overall. You have no possibility to switch back to the editor tab via accesskey so you have to use the mouse anyway.

It would be great if we could get this merged in [4.0.0](#) or [4.1.0](#).

#23 - 2018-02-03 03:27 - Mischa The Evil

Marius BALTEANU wrote:

[...] (using only the native accesskey attribute) [...]

Yeah, I consider them as pretty rudimentary ;)

Marius BALTEANU wrote:

- I would like to add some shortcuts for accessing the jstoolbar buttons (Control/Command + B for bold, Control/Command + I for italic, etc..).
- I will propose to use a JS library that implements a consistent shortcut key between the browsers. I find it very difficult to use the current implementation that depends on the browser and OS.

Both sound good, but especially the second is something that could help wide(r) adoption of accesskey use. The current dependency on browser and OS (by using the native attribute) is pretty horrific, if you'd ask me.

Though, I'd like to know first (as in before applying code that is knowingly breaking the preview accesskey functionality) if Jean-Philippe would accept any patches implementing accesskeys using JS at all.

Btw: issue [#6846](#) is related to this.

Marius BALTEANU wrote:

From my point of view, this missing accesskey should not be a blocker for this ticket.

I tend to agree in this case. However we should not forget that this particular accesskey has been available for over 10 years (since 0.6), so users could be very accustomed to it, making it a potentially hard change (like we've seen too with e.g. [#19468](#), which, partially, obsoleted the 'Start calendars on' setting).

Bernhard Rohloff wrote:

[...] the mostly unknown accesskey [...]

Well, I *think* this statement is mostly true, but we can't say for sure. The only fact that we know of is that the feature has been long available but that it never has been documented properly before [#27412](#) (it's not even covered in popular English books¹). I know Redmine admins, superusers and a small amount of managers (i.e. people who actually read the source code) who have been actively advocating to their users to make use of these accesskeys to up their productivity...

¹ like "Lesyuk, Andriy (2013). *Mastering Redmine*. Packt Publishing. ISBN [978-1-849519-14-4](#)", although I don't know if it changed with its second edition.

#24 - 2018-02-05 07:01 - Bernhard Rohloff

Mischa The Evil wrote:

Though, I'd like to know first (as in before applying code that is knowingly breaking the preview accesskey functionality) if Jean-Philippe would accept any patches implementing accesskeys using JS at all.

Yes, I agree. It would be great to have JP's opinion/vision on this topic.

I'm a heavy user of the shortcuts, too. And although I would miss this feature, I think this patch would gain the productivity of the majority of users who are not so familiar with the accesskeys. One can have a quick look at the preview without scrolling the page up and down. The editor always stays in its position which is also less distracting.

#25 - 2018-04-23 07:17 - Marius BĂLTEANU

@Go Maeda, is something missing from my patch in order to set the target version to [4.1.0](#)? I really want to see this change in the next major release.

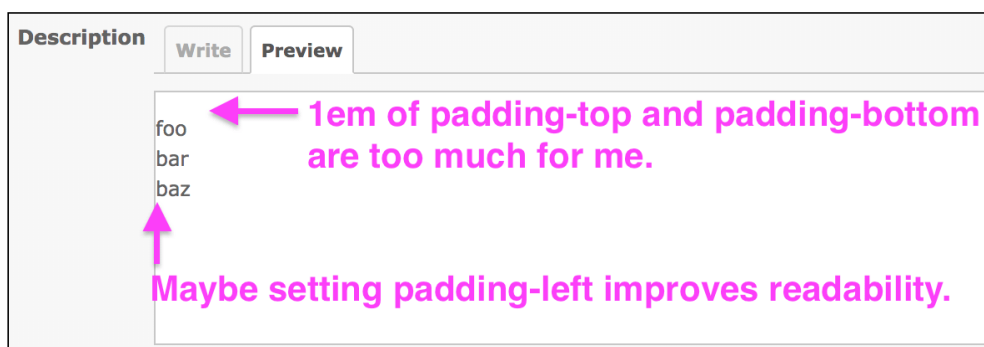
#26 - 2018-04-23 11:52 - Go MAEDA

- File *27758-padding@2x.png* added

- Target version changed from *Unplanned backlogs* to *Candidate for next major release*

Marius, thank you for working on this feature, I think it is a big improvement.

But there is one worrisome thing. Please see the following screenshot. I think we can improve padding value. 1em of padding before the first line is too much. And the left side of the paragraph is too close to the border on the left. What do you think?



#27 - 2018-04-23 21:32 - Marius BĂLTEANU

- File *preview_with_padding.png* added

Thanks for your feedback.

Now is better?

preview_with_padding.png

#28 - 2018-04-24 01:24 - Go MAEDA

Marius BALTEANU wrote:

Now is better?

preview_with_padding.png

It looks nice.

#29 - 2018-04-29 07:12 - Marius BĂLTEANU

- File *deleted (replace_preview_link_with_write_preview_tabs.patch)*

#30 - 2018-04-29 07:12 - Marius BĂLTEANU

- File *replace_preview_link_with_write_preview_tabs.patch added*

Updated the patch to include the fix suggested by Go Maeda.

#31 - 2018-04-29 10:15 - Go MAEDA

- File *write-20180429@2x.png added*

- File *preview2-20180429@2x.png added*

- File *preview1-20180429@2x.png added*

Marius BALTEANU wrote:

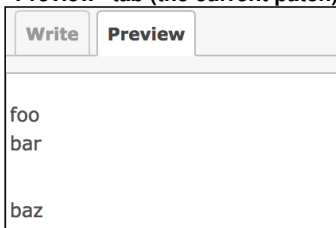
Updated the patch to include the fix suggested by Go Maeda.

Thanks. I think it will be perfect if the padding-top in the preview area is dropped. Please see the screenshots.

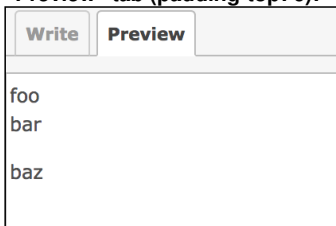
"Write" tab:



"Preview" tab (the current patch):



"Preview" tab (padding-top: 0):



```
.tabular .wiki-preview p {  
  min-height: initial;  
  padding: 1em 0 1em 0 !important;  
  overflow: initial;  
}
```


#32 - 2018-04-30 13:20 - Marius BĂLTEANU

- File `replace_preview_link_with_write_preview_tabs.patch` added

Attached the fixed version. Sorry for my previous patch, I've totally messed up the CSS rules (I wrote margin instead padding).

#33 - 2018-04-30 13:20 - Marius BĂLTEANU

- File deleted (`replace_preview_link_with_write_preview_tabs.patch`)

#34 - 2018-05-04 04:43 - Go MAEDA

- Assignee deleted (Marius BĂLTEANU)

- Target version changed from *Candidate for next major release* to *4.1.0*

Thank you for attaching the final patch. I suggest implementing this feature in the next release.

- [0001-drop-minified-version-of-jstoolbar-textile.patch](#)
- [0002-replace-preview-link-with-write-priview-tabs.patch](#)

#35 - 2018-07-28 14:02 - Martin Blom

Would love to see this patch included. It would make it much easier less hacky to handle previews in alternative editors like [Bestest Markdown Editor](#), which also integrates the preview view inside the editor.

Would it also add preview support for textarea elements that currently do not have a preview option, like when editing the issue description?

#36 - 2018-09-09 14:03 - Marius BĂLTEANU

- File deleted (`replace_preview_link_with_write_preview_tabs.patch`)

#37 - 2018-09-09 14:03 - Marius BĂLTEANU

- File deleted (`drop-jstoolbar-textile.min.patch`)

#38 - 2018-09-09 14:06 - Marius BĂLTEANU

- File `0001-drop-minified-version-of-jstoolbar-textile.patch` added

- File `0002-replace-preview-link-with-write-priview-tabs.patch` added

Updated the patches to apply cleanly on the current trunk and to prevent an error on IE browser.

Martin Blom wrote:

Would love to see this patch included. It would make it much easier less hacky to handle previews in alternative editors like [Bestest Markdown Editor](#), which also integrates the preview view inside the editor.

Would it also add preview support for textarea elements that currently do not have a preview option, like when editing the issue description?

Yes, it will work anywhere the toolbar is now rendered because the Write/Preview tabs are part of the toolbar.

#39 - 2018-09-25 04:32 - Go MAEDA

- Assignee set to *Jean-Philippe Lang*

- Target version changed from *4.1.0* to *4.0.0*

#40 - 2018-09-26 09:28 - Jean-Philippe Lang

- Status changed from *New* to *Closed*

- Resolution set to *Fixed*

Patch committed for 4.0. Marius, thank you for your work on this.

#41 - 2018-09-26 09:31 - Marius BĂLTEANU

Jean-Philippe Lang wrote:

Patch committed for 4.0. Marius, thank you for your work on this.

Thanks for committing it. I'll keep an eye on possible issues reported by the users.

#42 - 2018-09-26 09:37 - Jean-Philippe Lang

- Status changed from Closed to Reopened

Write and Preview are hardcoded strings in javascript. We have to support translations for these.

#43 - 2018-09-26 09:59 - Marius BĂLTEANU

I'm fixing the issue right now.

#44 - 2018-09-26 10:15 - Marius BĂLTEANU

I've looked in the code and support for translations already exists, please see line <source/trunk/public/javascripts/jstoolbar/jstoolbar.js#L101>. The jsTab function receives the tab name as a string and it checks first for translation. If exists, it uses the translated string, if not, it uses the string as tab name.

I think we need to add the new strings (Write, Previews) to all translations files. This is a manual process or there is a task to automatically add them?

#45 - 2018-09-26 10:16 - Wim DePreter

Jean-Philippe Lang wrote:

Write and Preview are hardcoded strings in javascript. We have to support translations for these.

Is "Edit" not better then "Write"?

#46 - 2018-09-26 10:30 - Go MAEDA

Marius BALTEANU wrote:

I think we need to add the new strings (Write, Previews) to all translations files. This is a manual process or there is a task to automatically add them?

Add keys to en.yml and run rake locales.

#47 - 2018-09-26 10:32 - Jean-Philippe Lang

Wim DePreter wrote:

Jean-Philippe Lang wrote:

Write and Preview are hardcoded strings in javascript. We have to support translations for these.

Is "Edit" not better then "Write"?

Yes, we can reuse Edit and Preview strings that are already present in locales.

#48 - 2018-09-26 10:38 - Jean-Philippe Lang

Marius BALTEANU wrote:

I think we need to add the new strings (Write, Previews) to all translations files. This is a manual process or there is a task to automatically add them?

I guess you're speaking about JS translations that sit in public/javascripts/jstoolbar/lang. There's no task to add strings to these. You can just update english and I'll take care of the others. Thanks.

#49 - 2018-09-26 11:26 - Marius BĂLTEANU

- File *replace_write_with_edit.patch* added

Here it is. I choose to replace also in the code Write with Edit.

#50 - 2018-09-26 13:02 - Jean-Philippe Lang

- Status changed from Reopened to Closed

I didn't see that the en locale was already filled in your original patch, thanks. I've updated all the others.

#51 - 2018-09-26 22:59 - Marius BĂLTEANU

- Related to deleted (Feature #7157: Content Editor Preview at the same place where text is)

#52 - 2018-09-27 18:06 - Marius BĂLTEANU

- Has duplicate Feature #7157: Content Editor Preview at the same place where text is added

#53 - 2018-09-30 06:06 - Takeshi Nakamura

- File toolbar-4.0.png added

- File toolbar-3.4.png added

Toolbar buttons are not wrapped when the screen is narrow.

Redmine 4.0

toolbar-4.0.png

Redmine 3.4

toolbar-3.4.png

#54 - 2018-09-30 06:11 - Go MAEDA

- Status changed from Closed to Reopened

- Assignee changed from Jean-Philippe Lang to Marius BĂLTEANU

Takeshi Nakamura wrote:

Toolbar buttons are not wrapped when the screen is narrow.

Marius, could you look into this?

#55 - 2018-09-30 09:19 - Marius BĂLTEANU

Go MAEDA wrote:

Takeshi Nakamura wrote:

Toolbar buttons are not wrapped when the screen is narrow.

Marius, could you look into this?

I know the issue, I've mentioned about this in note-8 ("on responsive mode, when the screen resolution width is less than the total width of the js toolbar (tabs + buttons), some buttons become hidden. I'm not happy with the current solution, but it is not a blocker from my point of view and we can improve this later when we have a better solution." but I'm not sure which should be the fix because the proposed solution will require a lot of space on small screens and also is not an easy fix.

A quick fix could be to hide the JS toolbar on small screens and on the long term, we can work to a better solution, for example:

- group some JS buttons under the same category: (eg: H1, H2 and H3 under H category)
- option to expand/collapse the JS buttons
- horizontal scroll without showing the scroll bar
- left/right buttons (as we have for menu)

IMHO, we can deliver this as it is and we'll fix it in the next minor release when for sure we will have other small improvements to be done for this feature.

#56 - 2018-09-30 11:01 - Go MAEDA

- Status changed from Reopened to Closed

- Assignee changed from Marius BĂLTEANU to Jean-Philippe Lang

Marius BALTEANU wrote:

IMHO, we can deliver this as it is and we'll fix it in the next minor release when for sure we will have other small improvements to be done for this feature.

I agree with you. Since it is a small issue, it should not delay the release of 4.0.0.

#57 - 2018-09-30 11:13 - Go MAEDA

- Related to Defect #29694: Some buttons in wiki toolbar are hidden in responsive layout added

#58 - 2018-10-07 02:37 - Go MAEDA

- Related to Feature #29712: Preview and wiki toolbar for full width custom fields added

#59 - 2019-04-02 07:56 - Marius BĂLTEANU

- Related to Defect #30988: Preformatted text overflows the preview area added

#60 - 2019-04-02 07:56 - Marius BĂLTEANU

- Related to Defect #30758: Preview URL in Wiki Toolbar should be escaped added

#61 - 2019-04-02 07:56 - Marius BĂLTEANU

- Related to Feature #30459: Switch edit/preview tabs with keyboard shortcuts added

#62 - 2020-12-14 03:18 - Go MAEDA

- Related to Patch #34444: Remove unused key :preview from Redmine::AccessKeys::ACCESSKEYS added

Files

write_preview.png	67.9 KB	2017-12-07	Marius BĂLTEANU
preview.png	66.6 KB	2017-12-07	Marius BĂLTEANU
write.png	68.9 KB	2017-12-11	Marius BĂLTEANU
preview.png	66.2 KB	2017-12-11	Marius BĂLTEANU
27758-many-lines.png	114 KB	2017-12-12	Go MAEDA
write.png	102 KB	2018-01-26	Marius BĂLTEANU
preview.png	76.1 KB	2018-01-26	Marius BĂLTEANU
27758-padding@2x.png	18.1 KB	2018-04-23	Go MAEDA
preview_with_padding.png	69.7 KB	2018-04-23	Marius BĂLTEANU
write-20180429@2x.png	8.83 KB	2018-04-29	Go MAEDA
preview2-20180429@2x.png	6.81 KB	2018-04-29	Go MAEDA
preview1-20180429@2x.png	6.81 KB	2018-04-29	Go MAEDA
0001-drop-minified-version-of-jstoolbar-textile.patch	11.6 KB	2018-09-09	Marius BĂLTEANU
0002-replace-preview-link-with-write-priview-tabs.patch	42.9 KB	2018-09-09	Marius BĂLTEANU
replace_write_with_edit.patch	2.12 KB	2018-09-26	Marius BĂLTEANU
toolbar-3.4.png	6.23 KB	2018-09-30	Takeshi Nakamura
toolbar-4.0.png	5.06 KB	2018-09-30	Takeshi Nakamura