

Issue recurring - task #27

More than one recurrence in the future

2020-07-07 12:04 - thwr

Status:	closed
Priority:	feature
Assignee:	cryptogopher
Category:	
Target version:	unspecified
Redmine version:	unspecified
Description	
First off, thanks for providing a <i>working</i> plugin with a well thought UI.	
During my tests, I've noticed that your plugin creates just one new issue in the future when running <code>renew_all</code> . It would be great to have a field to create more than one future issue, for example 4 or 10.	
Our use case is simple: We need to track people's time in the office during these challenging Corona times. For that, it would be good if people could create recurrences for up to three months in our case so that others could plan their office stays too, because we are not allowed to have more than one person in a usual office room for example.	
Related issues:	
Has duplicate task #5: Add configuration option to specify, how far into the ... rejected	

History

#1 - 2020-07-07 12:46 - cryptogopher

- Status changed from *new* to *in progress*
- Priority changed from *support* to *bug*

First off, thanks for providing a working plugin with a well thought UI.

Thank you for kind words. It is very rewarding to hear, that effect of my work is of any use for others. Actually the UI was my friend's idea, so I forwarded your words of appreciation :)

During my tests, I've noticed that your plugin creates just one new issue in the future when running `renew_all`. It would be great to have a field to create more than one future issue, for example 4 or 10.

To be strict: for recurrence schemes based on fixed schedule (i.e. those not depending on close time of last issue) the issues are created as long, as future issue reference date is *before tomorrow* and then one more future issue is created. If you run `renew_all` frequently enough it boils down to what you stated: there is always one future issue.

Our use case is simple: We need to track people's time in the office during these challenging Corona times. For that, it would be good if people could create recurrences for up to three months in our case so that others could plan their office stays too, because we are not allowed to have more than one person in a usual office room for example.

Given that you only care about recurrence schemes based on fixed schedule - that seems as a simple thing to do. I propose to add configuration setting which will tell at least how far into the future you want next issues to be created. For example: *0 days* would yield the same behavior as now, and *1 month* would create future issues up to 1 month from today *and* 1 additional issue after that . This way current logic would be maintained of always going one more recurrence further.

Regarding recurrence schemes based on closed date: they wouldn't be affected for obvious reasons.

Any comments/ideas regarding that?

Also this is the duplicate of [#5](#), so I'm closing the old one.

#2 - 2020-07-07 12:46 - cryptogopher

- Has duplicate task #5: Add configuration option to specify, how far into the future next recurrence issues are created added

#3 - 2020-07-07 13:54 - thwr

Thank you for kind words. It is very rewarding to hear, that effect of my work is of any use for others. Actually the UI was my friend's idea, so I forwarded your words of appreciation :)

You're welcome. I'm a developer too and know very well how much appreciation goes when it comes to software. Like

```
love < void
```

:)

To be strict: for recurrence schemes based on fixed schedule (i.e. those not depending on close time of last issue) the issues are created as long, as future issue reference date is before tomorrow and then one more future issue is created. If you run `renew_all` frequently enough it boils down to what you stated: there is always one future issue.

Got it, thanks.

Given that you only care about recurrence schemes based on fixed schedule - that seems as a simple thing to do. I propose to add configuration setting which will tell at least how far into the future you want next issues to be created. For example: 0 days would yield the same behavior as now, and 1 month would create future issues up to 1 month from today and 1 additional issue after that . This way current logic would be maintained of always going one more recurrence further.

Regarding recurrence schemes based on closed date: they wouldn't be affected for obvious reasons.

Any comments/ideas regarding that?

I think we are on the right track here. Thank you very much.

Also this is the duplicate of [#5](#), so I'm closing the old one.

Oh, sorry, didn't see [#5](#). My bad.

#4 - 2020-07-14 22:16 - cryptogopher

- Status changed from in progress to closed

Feature has been added. You can check it out now using

```
git checkout master
```

(and executing other steps according to upgrade instructions) or wait until next release.

After upgrade you can set renew ahead period in plugin settings.

#5 - 2020-07-15 08:00 - thwr

cryptogopher wrote:

Feature has been added. You can check it out now using
[...]
(and executing other steps according to upgrade instructions) or wait until next release.

After upgrade you can set renew ahead period in plugin settings.

Awesome, thank you. Will check that out and report back.

#6 - 2020-07-26 00:00 - thwr

Sorry for my late feedback. I've tried your commit https://github.com/cryptogopher/issue_recurring/commit/4bed4ed02737bb25aa03f1c67854858331ca27b0 for a few days now and I can't say much here... Everything works great, can't see any failures. We've created roughly 500 tickets in the past days with your plugin.

Thank you for your good work and for the fast release of my feature request. Really appreciate it.

#7 - 2020-07-26 00:05 - cryptogopher

thwr wrote:

Thank you for your good work and for the fast release of my feature request. Really appreciate it.

You're welcome. Good to hear everything works fine :)

#8 - 2023-06-08 15:22 - admin

- *Priority changed from bug to feature*

#9 - 2023-06-08 15:31 - admin

- *Tracker changed from 2 to task*

- *Target version set to unspecified*

- *Redmine version set to unspecified*