

Calculate %done based on estimated and logged time

2022/05/09 16:45 - Admin Redmine

ステータス:	New	開始日:	2022/05/09
優先度:	通常	期日:	
担当者:		進捗率:	0%
カテゴリ:	Issues planning_40	予定工数:	0.00時間
対象バージョン:	Candidate for next minor release_33	作業時間:	0.00時間
Redmineorg_URL:	https://www.redmine.org/issues/13400	status_id:	1
category_id:	40	tracker_id:	3
version_id:	33	plus1:	0
issue_org_id:	13400	affected_version:	
author_id:	2784	closed_on:	
assigned_to_id:	0	affected_version_id:	
comments:	20		

説明

One of our customers at "Planio":<http://plan.io/redmine-hosting/> requested to have the %done field calculated based on estimated and logged time. This patch is extracted from the Planio code base.

If enabled, it calculated the percent done according to these rules:

- if estimated time is set: @logged_time / estimated_time@ ** the number is never higher than 100%, even if @logged_time > estimated_time@
- if @estimated_time@ is not set: @0@

Logged and estimated times of child issues are taken into account, similarly to the way it is done when using the issue status. If a user logs/modifies/deletes time, it creates a journal entry of the changed %done on the affected issue. It will not create journals on parent issues that might also be updated.

The patch is attached.

journals

Dupe of #12762?

Jan Niggemann wrote:

Dupe of #12762?

I'd say just related since this is a patch which implements feature #12762.

Could you use rather use @Issue@ model @time_entries@ association @after_add@ and @after_remove@ callbacks rather than @TimeEntry@ model ones?

Etienne Massip wrote:

Could you use rather use @Issue@ model @time_entries@ association @after_add@ and @after_remove@ callbacks rather than @TimeEntry@ model ones?

Well, you can add/edit/delete time entries all without even touching an issue. I'm not sure how that is supposed to work as in this case, the issues isn't even loaded.

Also one tiny remark which I forgot above: the patch is against Redmine 1.4.7. If this is

something that should go into the core, I could probably port that to 2.2 (haven't tested it there).

Holger Just wrote:

Also one tiny remark which I forgot above: the patch is against Redmine 1.4.7. If this is something that should go into the core, I could probably port that to 2.2 (haven't tested it there).

I think it would have its place in core so it would be great to have this patch against a recent trunk revision.

Holger Just wrote:

Etienne Massip wrote:

Could you use rather use @Issue@ model @time_entries@ association @after_add@ and @after_remove@ callbacks rather than @TimeEntry@ model ones?

Well, you can add/edit/delete time entries all without even touching an issue. I'm not sure how that is supposed to work as in this case, the issues isn't even loaded.

I didn't think that the actual @TimeEntry@ model was this much tied to the issue the time entry belongs to, looking a bit more deeply in its code, it's very OK like you did.

FTR: related and duplicating issues are listed in [\[\[DoneRatio_issues_overview\]\]](#).

attachment:0001-patch.diff seems for trunk r9702.

I removed languages except en and de.

This is a patch rebased to trunk r12075.

I applied the patch issue-13400-trunk-r12075.diff on Redmine 2.4.2.stable with success but I get that error when submitting an issue:

```
NameError (undefined local variable or method `update_done_ratio_from_issue_status' for #):
  app/models/issue.rb:165:in `create_or_update'
  app/controllers/issues_controller.rb:470:in `block in save_issue_with_child_records'
  app/controllers/issues_controller.rb:458:in `save_issue_with_child_records'
  app/controllers/issues_controller.rb:184:in `update'
```

Is there any other solution to get this feature on 2.4.2 ?

Vincent Adelé wrote:

I applied the patch issue-13400-trunk-r12075.diff on Redmine 2.4.2.stable with success but I get that error when submitting an issue:
[...]

Is there any other solution to get this feature on 2.4.2 ?

I have this problem on 3.0.2 and I need this feature

I recently came across this issue while working at Planio. I've rebased Holger's patch and added some tests.

It would be great, if this feature suggestion could be reconsidered.

Thank you Gregor for working on this. I think this is a great addition.

I'm setting the version to Candidate for next minor release, especially in the light of [\[\[DoneRatio issues overview\]\]](#).

Screenshots of attachment:0001-Calculate-done_ratio-based-on-logged-time.patch.

!{width: 560px; border: 1px solid #ccc}.settings_issue_done_ratio@2x.png!

!{width: 698px; border: 1px solid #ccc}.calculated-done-ratio@2x.png!

There are many similar requests: [#952](#), [#3719](#), [#7545](#), [#11999](#), [#12762](#), [#27965](#)

I would like to discuss the implementation of [#24277](#) instead of this feature because I found it more complete and flexible for the users and also, it incorporates this functionality by default.

I think the proposed feature is appropriate to say "% spent" rather than "% done". And it shows strange "% done" value for closed issues in some circumstances. Suppose that a person finished an issue in 5 hours, and the issue was estimated to take 10 hours. The person finished the issue ahead of schedule but the closed issue shows that "% done" is 50%.

Thank you for your feedback:

@Marius BALTEANU: Thank you for pointing out [#24277](#). I think it's a nice addition. My main goal was to keep this issue here alive, so that people are able to use the feature eventually.

@Go MAEDA: I think, one may argue both ways. Yes it's weird, that the issue is less than 100 % done, but closed already. On the other hand, this may be seen as a simple indicator, that you've stayed below budget.

In the end, I think Redmine should be consistent. Since the fields are also not set to 100 % when closing an issue with the "use the issue field" setting, I think it's expected to handle it the same way with this new setting. The status is only taken into account, when selecting "Use the issue status".

related_issues

relates,New,12762,Add option to calculate done ratio with the data from time tracking

relates,New,24277,Introducing Remaining Time field as method to track the remaining time to complete an issue

履歴

#1 - 2022/05/10 17:13 - Admin Redmine

- カテゴリ を Issues planning_40 にセット

- 対象バージョン を Candidate for next minor release_33 にセット