

Wrong formatting of date in Time Entries

2022/05/09 19:22 - 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:	Time tracking_13	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/36897	status_id:	1
category_id:	13	tracker_id:	3
version_id:	33	plus1:	0
issue_org_id:	36897	affected_version:	
author_id:	582905	closed_on:	
assigned_to_id:	0	affected_version_id:	
comments:	7		
説明			
Date is formatted wrong at " – Time entries" ("7:60" instead of "8:00", for example). See screenshot.			
This patch fixes it.			
journals			
I've encountered this in 4.0.6, but AFAIK this bugged part of code still not fixed in 5.0.0.			
This whole thing can probably be simplified to just			
<pre>def format_hours(hours) return "" if hours.blank? if Setting.timespan_format == 'minutes' "%d:%02d" % (hours * 60).divmod(60) else l_number("%0.2f", hours.to_f) end end</pre>			
Probably yes.			
But as far as I understood from how hours are displayed in Time entries, they need to be .round-ed (not just dropping the fractional part).			
I.e. if you've spent 7:59:30 (7.991666666666667 hrs), for human it's 8:00 (we don't count seconds in Time entries). :)			
Patch gives "8:00" too.			
I think like this.			
Hmmm, with my version, trailing "seconds" indeed would just be cut off instead of being rounded. Thus, semantically @7:59:01@ and @7:59:59@ would both be shown as @"7:59"@ which is different from the current behavior.			
If we round the minutes, we would again have to deal with the case of @7.99@ hours being rounded up to @"7:60"@ however.			
Maybe this version could work then:			
<pre>h, m = (hours * 60).divmod(60).map!(&:round) if m == 60 h += 1</pre>			

```
    m = 0
end
"%d:%02d" % [h, m]
```

The @if@ block could also be shortened to the equivalent one-liner:

```
h, m = h+1, 0 if m == 60
```

In any case, we can be sure that after the first line, @m@ is an Integer between 0 and 60 (inclusive). If it was rounded up to @60@, we set to to @0@ and increase the hour by one in the @if@ block.

This should then be equivalent to the proposed version in the original patch, but with a little less math involved and a little bit more straight-forward.

At Planio, we run this code for some time now which is a mixture of both approaches:

```
h = hours.floor
m = ((hours - h) * 60).round
if m == 60
  h += 1
  m = 0
end
"%d:%02d" % [h, m]
```

Here, the first two lines (which are the same as in Redmine core) return the exact same result as @h, m = (hours * 60).divmod(60).map!(&:round)@.

Holger Just wrote:

At Planio, we run this code for some time now which is a mixture of both approaches:

```
h = hours.floor
m = ((hours - h) * 60).round
if m == 60
  h += 1
  m = 0
end
"%d:%02d" % [h, m]
```

Here, the first two lines (which are the same as in Redmine core) return the exact same result as @h, m = (hours * 60).divmod(60).map!(&:round)@.

Yes, I think we can go this way. :)

So, the adjusted patch.

related_issues

relates,Closed,23996,Introduce a setting to change the display format of timespans to HH:MM

履歴

#1 - 2022/05/10 16:57 - Admin Redmine

- カテゴリを Time tracking_13 にセット

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