Redmine - Defect #36897
Wrong formatting of date in Time Entries
2022-04-05 16:04 - Denis Yurashku

Date is formatted wrong at "<PROJECT> – Time entries" ("7:60" instead of "8:00", for example). See screenshot.

This patch fixes it.

Related issues:
Related to Redmine - Feature #23996: Introduce a setting to change the displa... Closed

History
#1 - 2022-04-05 16:30 - Denis Yurashku
I've encountered this in 4.0.6, but AFAIK this bugged part of code still not fixed in 5.0.0.

#2 - 2022-04-06 15:21 - Holger Just
- Related to Feature #23996: Introduce a setting to change the display format of timespans to HH:MM added

#3 - 2022-04-06 15:23 - Holger Just
- Target version set to Candidate for next minor release

This whole thing can probably be simplified to just

```ruby
def format_hours(hours)
  return "" if hours.blank?
  if Setting.timespan_format == 'minutes'
    "%d:%02d" % (hours * 60).divmod(60)
  else
    l_number("%.2f", hours.to_f)
  end
end
```

#4 - 2022-04-06 16:39 - Denis Yurashku
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.

#5 - 2022-04-06 17:35 - Holger Just
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:

```ruby
h, m = (hours * 60).divmod(60).map(&:round)
if m == 60
  h += 1
  m = 0
```

2024-05-20 1/4
The if block could also be shortened to the equivalent one-liner:

\[
\text{h, m = h+1, 0 if m == 60}
\]

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 it 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 straightforward.

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

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

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

#6 - 2022-04-20 14:42 - Denis Yurashku

Holger Just wrote:

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

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

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

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

Depending on the database and the rounding of the floats involved, this situation may also be created with other times as shown in the screenshot as we always round the raw values to two decimal places before summing them in TimeEntryQuery#total_for_hours which may introduce additional error margins.

Holger Just wrote:

You can reproduce the issue by creating a time entry with decimal logged hours slightly below 8 hours, e.g. 7.991666666666667 hours. Here, the fractional minutes will be rounded up to 60 minutes because 0.991666666666667 * 60 = 59.50000000000002.

I have confirmed the issue. The spent time value 7.991666666666667 is displayed as "7:60 h" on the issue page and displayed as "8:00 h" after applying the patch.

But I am not sure if the value should be displayed as "8:00 h" there because it is displayed as "7:59" on the "Spent time" tab. The reason why you see different values on each page is that the Spent time page uses a value 7.99 rounded by TimeEntry#hours (source:tags/5.0.2/app/models/time_entry.rb#L196), while Issue page uses a value 7.991666666666667 obtained from the database (source:tags/5.0.2/app/models/issue.rb#L1140) without using TimeEntry#hours.

---

**Bug #18**

*OPEN*

---

**Status:** New  
**Start date:** 07/19/2022

**Priority:** Normal  
**Due date:**

**Assignee:**  
**% Done:**

**Category:**  
**Estimated time:**

**Target version:**  
**Spent time:** 8:00 h

**Searchable field:** Default string  
**Custom date:**

**Database:**  
**Project 1 cf:**

---

**Spent time**

**Options**

**Details**  
**Save custom query**

---

**Files**

<table>
<thead>
<tr>
<th>File</th>
<th>Size</th>
<th>Date</th>
<th>Author</th>
</tr>
</thead>
<tbody>
<tr>
<td>Format_hours.diff</td>
<td>450 B</td>
<td>2022-04-05</td>
<td>Denis Yurashku</td>
</tr>
<tr>
<td>Снимок экрана от 2022-04-05 16-59-25_0.png</td>
<td>71,6 KB</td>
<td>2022-04-05</td>
<td>Denis Yurashku</td>
</tr>
</tbody>
</table>