Redmine - Defect #42339

Value for "hours" incorrect when creating a time entry using the API

2025-02-28 16:04 - Stefan Borchert

Status:	Closed		**Start date:**	
Priority:	Normal		**Due date:**	
Assignee:			**% Done:**	0%
Category:	Time tracking		**Estimated time:**	0.00 hour
Target version:				
Resolution:	Wont fix		**Affected version:**	6.0.3

Description

Description

We use Redmine version 6.0.3.stable and a custom script to synchronize time entries from *toggl* to Redmine.
This worked well so far but after the update to 6.0.3.stable we noticed that some time entries have incorrect values for hours.
Example: we are creating a new time entry and send "hours: 0.44". The resulting time entry has the value "0.43" for property "hours".

Is there any sanitizing or rounding of values in Redmine weare unaware of? It worked in Redmine 5 so there has to be some changes.

Environment

```
Redmine version              6.0.3.stable
  Ruby version               3.2.3-p157 (2024-01-18) [x86_64-linux-gnu]
  Rails version              7.2.2.1
  Environment                production
  Database adapter           PostgreSQL
  Mailer queue               ActiveJob::QueueAdapters::AsyncAdapter
  Mailer delivery            sendmail
```

Related issues:

Related to Redmine - Defect #40914: Fix precision issues in TimeEntry#hours c...	**Closed**

History

#1 - 2025-02-28 21:21 - Stefan Borchert

- File 42339-01.mp4 added

I've tested creating the time entry using postman and got the same result (see attached video). Creating a time entry with <hours>0.44</hours> results in a new item with <hours>0.43</hours>.
Using other values works.

#2 - 2025-03-01 08:17 - Go MAEDA

- Category changed from REST API to Time tracking

Since Redmine 6.0.0, decimal values entered in the "Spent time" field are rounded to the nearest minute. For example, 0.44 hours (26 minutes and 24 seconds) is rounded to 26 minutes and stored in the database as 0.43333333333333335. When displayed on the screen, it is shown with two decimal places as 0.43.

This change was made in [#40914](#) to prevent calculation errors in spent time.

#3 - 2025-03-01 10:51 - Stefan Borchert

Thanks for the info. I'll update our script accordingly to round the values differently.

#4 - 2025-03-02 02:12 - Go MAEDA

- Status changed from New to Closed

- Resolution set to Wont fix

Thank you for your feedback. I am closing this issue.

#5 - 2025-03-02 02:20 - Go MAEDA

- Related to Defect #40914: Fix precision issues in TimeEntry#hours calculation by returning Rational instead of Float added

Files

42339-01.mp4	391 KB	2025-02-28	Stefan Borchert