

Redmine - Feature #35764

Multiple search terms in the "contains" operator of text filters

2021-08-15 07:22 - Go MAEDA

Status: Closed	Start date:
Priority: Normal	Due date:
Assignee: Marius BALTEANU	% Done: 0%
Category: Filters	Estimated time: 0.00 hour
Target version: 5.0.0	
Resolution: Fixed	

Description

I extracted the attached patch from #35073#note-4. The patch posted by [Jens Krämer](#) from Planio allows you to use multiple search terms in "contains" operator of text filters.

The alternative I am experimenting with right now is breaking up the user's query string into tokens like the global search does, and use these to build a query with multiple LIKE clauses that are combined with AND. I'm attaching a preliminary patch that implements this for the Issue.like scope (used by the autocompleter) and the Query#sql_contains method (which should cover all query filters). Do you think that's a viable approach?

It seems that the patch has already been deployed to the production environment of Planio. See [Planio filters just got more powerful - Planio Support (english) - Planio Support](#)

Related issues:

Blocked by Redmine - Patch # 35073: Escape values in LIKE statements to preve... **Closed**

Associated revisions

Revision 21238 - 2021-10-05 21:54 - Marius BALTEANU

Tokenize search parameter in order to allow multiple search terms in:

- the "contains" operator of text filters
- in issue autocomplete

Patch by Jens Krämer.

Revision 21239 - 2021-10-05 21:55 - Marius BALTEANU

Add tests for #35764.

Revision 21241 - 2021-10-09 08:47 - Marius BALTEANU

Disable rubocop Lint/IneffectiveAccessModifier for self.tokenized_like_conditions in app/models/query.rb (#35764).

History

#1 - 2021-08-15 07:23 - Go MAEDA

- Blocked by Patch #35073: Escape values in LIKE statements to prevent injection of placeholders (_ or %) added

#2 - 2021-08-15 10:12 - Marius BALTEANU

- Target version set to 5.0.0

#3 - 2021-10-03 22:03 - Marius BALTEANU

- Assignee set to Marius BALTEANU

#4 - 2021-10-05 22:09 - Marius BALTEANU

- Resolution set to Fixed

Patch committed with tests, thanks for this nice improvement.

The patch introduces the following offense:

Lint/IneffectiveAccessModifier: private (on line 1074) does not make singleton methods private. Use private_class_method or private inside a class << self block instead.

```
def self.tokenized_like_conditions(db_field, value, **options)
```

#5 - 2021-10-09 08:50 - Marius BALTEANU

- Status changed from New to Resolved

I've disabled for now the rubocop offense for Lint/IneffectiveAccessModifier. Jens, the method self.tokenized_like_conditions should not be public instead of private?

#6 - 2021-10-09 13:12 - Marius BALTEANU

- Status changed from Resolved to Closed

Files

0001-tokenize-query-strings-for-Issue.like-and-Query-sql_.patch	3.93 KB	2021-08-15	Go MAEDA
---	---------	------------	----------