

Redmine - Feature #5840

Live lookup of the existing issues based on the keywords entered in the title for new issue

2010-07-08 15:43 - Andrius Kriučkovas

Status: New	Start date: 2010-07-08
Priority: Normal	Due date:
Assignee:	% Done: 50%
Category: UI	Estimated time: 0.00 hour
Target version: Candidate for next major release	
Resolution:	
Description It would be good benefit for duplicate issue evasion, to have live look-up of the existing issues based on the keywords entered in the title. Simply, as you enter the keywords in the title of a new issue, system scans DB for any issues and offers for example up to 5 top-related issues, which have most of the keywords in the title. This would greatly improve duplication problem, albeit putting strain on the server. This feature should not try to look-up on any character change in the title, but only after the full word is entered and space symbol following it.	
Related issues:	
Related to Redmine - Feature #9180: Improve search system for issues - like "...	New 2011-09-04
Related to Redmine - Feature #5492: Issues should implement acts_as_ferret	New 2010-05-10
Related to Redmine - Feature #5470: Avoid duplicate issue creation	New 2010-05-07
Related to Redmine - Feature #31989: Inline issue auto complete (#) in fields...	Closed
Has duplicate Redmine - Feature #5859: Search for similiar tickets when creat...	Closed 2010-07-09
Has duplicate Redmine - Feature #9642: Duplicate issue search on new issue su...	Closed 2011-11-23
Has duplicate Redmine - Feature #12240: Check for duplicate/related tickets b...	Closed

History

#1 - 2010-07-09 15:31 - Aron Rotteveel

+1. Would love to see a feature like this.

#2 - 2010-07-09 15:38 - Felix Schäfer

- *Category changed from Search engine to UI*

This should be a breeze to throw in once we have a unified autocomplete mechanism. What do you propose should happen on clicking on the matched issues?

#3 - 2010-07-09 16:59 - Andrius Kriučkovas

Felix Schäfer wrote:

This should be a breeze to throw in once we have a unified autocomplete mechanism.

Sorry, I am not familiar with the development roadmap, therefore, I cannot comment on the architecture.

What do you propose should happen on clicking on the matched issues?

I think, if the mouse-cursor over of those top-5 issues, should show the header of the problem on the pop-up section (or whatever it is in the HTML terms) with the main description.

And clicking it - go directly to the old issue, instead of post new one. We encourage the users to search for old issues, and reopen them, but this is not trivial and requires active involvement. Such feature would kill the issue in a smart way.

#4 - 2011-01-21 12:25 - Andrew Betts

This would be very useful for us as well. Our Redmine users include our clients as well as our development team and on commercial projects you cannot demand that a client go out of their way to try and find out whether the issue they've observed has already been reported in the same way that you might expect an open source user to. The particular case I'm thinking of is a client that has small offices in five locations around the world. The staff don't regularly see each other and are incredibly busy so collaboration on issue tracking is currently virtually non-existent.

Having the ability to pro-actively let them know that the issue they're raising has already been reported, and giving them a link to it, would save a lot of

time and improve communication and co-ordination, particularly over urgent issues.

#5 - 2011-01-26 21:38 - Andrew Betts

- File Capture.PNG added

I've solved this for us with a bit of jQuery. This clearly isn't the way you'd do it ideally, but I'm a JS developer, and I don't know any Ruby :-)

```
// When new issues are created, search Redmine for similar issues
var timSearchSimilar = null;
$('.new-issue-form #issue_subject').attr('autocomplete', 'off').bind('keyup change', function() {
  var txtEl = $(this);
  clearTimeout(timSearchSimilar);
  timSearchSimilar = setTimeout(function() {
    var proj = location.href.replace(/^.*/projects\/([\^\/]+)\issues.*$/, '$1');
    $.get('/search/index/'+proj+'?issues=1&q='+escape(txtEl.val()), function(resp) {
      var pat = new RegExp(/&lt;a href="\//issues\/(\d+)\&gt;(.*) #(\d+) \((([\^\/]+)\)\: (.+?)</a>/gi);
      var result;
      var similarhtml = '';
      while ((result = pat.exec(resp)) != null) {
        similarhtml = "&lt;tr&gt;&lt;td&gt;"+result[2]
        "&lt;/td&gt;&lt;td&gt;&lt;a href='/issues/'+result[1]+'&gt;#+result[1]&lt;/a&gt;&lt;/td&gt;&lt;td&gt;"+result[5]+&lt;/td&gt;&lt;td&gt;"+result[4]+&lt;/td&gt;&lt;/tr&gt;";
      }
      if ($('#similarissues').length) $('#similarissues').remove();
      if (similarhtml) {
        similarhtml = "&lt;div id='similarissues' style='padding: 5px 0 8px 180px;'&gt;&lt;label&gt;Did you mean?&lt;br&gt;&lt;span style='font-size:10px; line-height: normal; font-weight: normal'&gt;These issues already exist&lt;/span&gt;&lt;/label&gt;&lt;table style='line-height: 1em'&gt;"+similarhtml+"&lt;/table&gt;&lt;/div&gt;";
        $('#parent_issue').before(similarhtml);
      }
    }, 500);
  });
});
```

I hope this is of some use to people who want this feature until it's available in Redmine itself.

Result looks like this:

Capture.PNG

#6 - 2011-01-26 21:40 - Andrew Betts

Note that in my previous comment, the JS code doesn't have any HTML entities in it, so convert those back to angle brackets for the real code.

#7 - 2011-02-07 22:04 - Dustin Lambert

- File dups.js added

I like Andrew's solution, but in the spirit of Prototype, I've translated his solution to Prototype. It should work with Redmine out of the box as a plugin.

Also changed a few things (I hope):

- Changed regular expression to handle Redmine being installed to a sub directory
- Open links in new windows
- Tracker Type and Issue Number are all linked now

Hope this helps!

#8 - 2011-03-30 11:10 - Anonymous

Where do i have to put the dups.js? How do i integrate it into redmine?
Thanks in advance

#9 - 2011-04-23 10:30 - Andrius Kriučkovas

- % Done changed from 10 to 50

Would it be possible to release this into mainline and/or as plug-in? From end-user POV, there is lack of install instructions.

#10 - 2011-11-24 12:39 - David B

+1 I want this too!

#11 - 2011-11-24 13:56 - Etienne Massip

- Target version set to Candidate for next major release

#12 - 2011-11-24 16:42 - Terence Mill

+1

#13 - 2011-11-24 22:20 - Mischa The Evil

- File ro5840-integrate_dups_js_by_dustin_lambert-trunk_r7885.patch added

I've created a simple patch file to integrate the dups.js contributed by Dustin Lambert within the core. I haven't touched the javascript but it seems to work without any problems (haven't tested it thoroughly though).

The here provided implementation can also be provided through a plugin pretty easily. I think the following two topics should be covered now:

1. We should reach consensus about if this should be integrated within the Redmine core or through a third-party Redmine plugin (unless this is already decided by the committers, considering the target-version)
2. If the decision is made to integrate this as a core feature, I'd prefer to have an option to enable/disable this feature preferably at project level but at least at application level. Any opinions?

#14 - 2011-11-24 23:08 - Jean-Philippe Lang

- Target version changed from Candidate for next major release to Unplanned backlogs

I'm not sure that this should be in the core. The proposed dup.js seems more like a hack to me but can easily be added via a plugin.

#15 - 2011-11-24 23:13 - Jean-Philippe Lang

BTW, to make this script more robust, the core could provide json/xml response for search results.

#16 - 2011-11-25 08:46 - Andrius Kriučkovas

Jean-Philippe Lang wrote:

I'm not sure that this should be in the core. The proposed dup.js seems more like a hack to me but can easily be added via a plugin.

Dear Jean. From the support point of view, the plug-in is worse. For example in Lithuania, many supporters of the Redmine decline to mess up with the plugins, especially if they lag-behind the upgrade cycle/version. I would like it to suggest to be a core feature, because it truly think, that this would have a very positive impact of the real implementations in the field - both for knowledge search and duplication evasion.

Of course the impact on the DB and server-side resources should be accounted for, but I think this is not an issue any more because of move to 64 bits/more RAM, many-cores, etc.

#17 - 2011-11-25 08:50 - Andrius Kriučkovas

One more thing - there is an issue with the languages for example (Lithuanian), which have many suffixes, endings, and which are not only root-based (as English is). Declensions and conjugations does have impact.

I think I should update initial report, that the system should try to look-up after N (configurable via settings) symbols are entered and after space (or short pause after last input character). This would almost solve such language issues.

#18 - 2011-11-25 23:50 - Jean-Philippe Lang

Vasaris Vėjas wrote:

From the support point of view, the plug-in is worse.

I understand but I simply can't put all the existing plugins in the core.

I think I should update initial report, that the system should try to look-up after N (configurable via settings) symbols are entered and after space (or short pause after last input character). This would almost solve such language issues.

That's the point that makes this feature not so easy to make it **really usefull**. This could work if it relies on a smarter full text search engine (eg. tsearch for pg).

#19 - 2011-11-26 08:40 - Andrius Kriučkovas

Jean-Philippe Lang wrote:

Vasaris Vėjas wrote:

From the support point of view, the plug-in is worse.

I understand but I simply can't put all the existing plugins in the core.

At least for now I hope ;-)

That's the point that makes this feature not so easy to make it **really usefull**. This could work if it relies on a smarter full text search engine (eg. tsearch for pg).

I do agree, that this is not obviously a killer-feature. But I do hope, that in time it will make to a supported plug-in or core. My point was, that even not a perfect solution (80%:20%) if looked from the end-user/supporting admin (not developer's!) position would be useful. At least a candidate for a consideration ;-).

P.S. By the way I think that most latin-based languages wouldn't have any issues with the simple implementation.

#20 - 2012-02-05 14:24 - Alessandro Bahgat

Hi guys, we started working on a plugin (here https://github.com/abahgat/redmine_didyemean) to provide this kind of functionality, and I just discovered this issue. We are still experimenting with a few search options in order to return just relevant results and, ideally, we'd love to do it without impacting Redmine's core.

If you want to help with any hint or suggestion, we'd really appreciate that.

#21 - 2013-02-01 09:22 - Terence Mill

+1

Covered by:

Feature [#9180](#)- Improve search system for issues - like "context specific search"

#22 - 2013-02-08 09:25 - Daniel Felix

This seems to be covered by <http://www.redmine.org/plugins/didyemean> ?

Edit: Haven't seen note 20. :-)

But maybe you could provide this as a corepatch to add this in Redmine core in the future? :-)

#23 - 2013-04-04 15:00 - Andrius Kriučkovas

- Status changed from New to Resolved

Alessandro Bahgat wrote:

Hi guys, we started working on a plugin (here https://github.com/abahgat/redmine_didyemean) to provide this kind of functionality, and I just discovered this issue. We are still experimenting with a few search options in order to return just relevant results and, ideally, we'd love to do it without impacting Redmine's core.

If you want to help with any hint or suggestion, we'd really appreciate that.

I have just installed this one, as my scheduled Redmine upgrade sequence. From the first sight, it looks exactly what was needed. Perhaps it would be nice, if Redmine had some search tags infrastructure, so the live lookup would work even for tags/contents of the issue. Because users sometimes are laconic on the subject line ;-)

#24 - 2013-04-04 15:02 - Andrius Kriučkovas

Daniel Felix wrote:

This seems to be covered by <http://www.redmine.org/plugins/didyemean> ?

I bet, it is.

Edit: Haven't seen note 20. :-)

But maybe you could provide this as a corepatch to add this in Redmine core in the future? :-)

As plugin install takes ~3 min., it is no longer an issue.

#25 - 2013-04-04 15:39 - Etienne Massip

- Status changed from Resolved to New

#26 - 2013-06-26 05:39 - Jeff Pierson

+1 This is a feature that Bugzilla had that we appreciated greatly once it was added. We will be soon migrating to Redmine and likewise having duplicate issue detection at some point in the future would be a very welcome feature.

#27 - 2015-06-24 12:52 - Jaap de Haan

+1: StackOverflow has a feature like this and it's really great. Unfortunately not open source so no details on implementation available but that's the way it should work, definitely :-)

#28 - 2018-09-17 15:25 - Anonymous

+1

#29 - 2018-09-25 09:40 - Nadav Kavalерchik

+1

#30 - 2018-09-25 11:19 - Nadav Kavalерchik

I have tested: https://github.com/abahgat/redmine_didyemean

but it does not support Redmine version 3+

And since I am not a Ruby developer, I preferred the JS hack kindly shared above by Andrew Betts, 7 years ago!

A working solution for Redmine 3.4.6 stable

I have installed the plugin:

https://github.com/martin-denizet/redmine_custom_js

And updated the following JS code to support latest stable Redmine 3.4

```
$(document).ready(function() {
  // Wait for page to load - start

  // When new issues are created, search Redmine for similar issues
  var timSearchSimilar = null;
  $('new_issue #issue_subject').attr('autocomplete', 'off').bind('keyup change', function() {
    var txtEl = $(this);
    clearTimeout(timSearchSimilar);
    timSearchSimilar = setTimeout(function() {
      var proj = location.href.replace(/^.*/projects\/([\^\/]+)\.issues.*$/, '$1');
      $.get('/projects/'+proj+'/search?utf8=&&scope=&issues=1&q='+txtEl.val(), function(resp) {
        var pat = new RegExp(/<a href="\//issues\/(\d+)\>(.*?) #(\d+) \(([\^\/]+)\): (.+?)</a>/gi);
        var result;
        var similarhtml = '';
        while ((result = pat.exec(resp)) != null) {
          similarhtml = similarhtml + "<tr><td>" + result[2] + "</td><td><a href='/issues/' + result[1] + '">" + result[1] + "</a></td><td>" + result[5] + "</td><td>" + result[4] + "</td></tr>";
        }
        if ($('#similarissues').length) $('#similarissues').remove();
        if (similarhtml) {
          similarhtml = "<div id='similarissues' style='padding: 5px 0 8px 180px;'>" +
            "<label>Did you mean?<br><span style='font-size:10px; line-height: normal; font-weight: normal;'>These issues already exist:</span></label>" +
            "<table style='line-height: 1em;'>" + similarhtml + "</table></div>";
          $('#issue_subject').after(similarhtml);
        }
      });
    }, 500);
  });

  // Wait for page to load - end
});
```

#31 - 2018-11-09 09:00 - Kai Forberger

+1: The solution from [#5840-30](#) works for us. It would be great to have a duplicate issue detection feature in redmine.

#32 - 2019-09-01 06:13 - Go MAEDA

- Related to Feature #31989: Inline issue auto complete (#) in fields with text-formatting enabled added

#33 - 2019-10-03 10:29 - Takenori TAKAKI

- File *live_look_up_similar_issues.png* added

- File *live_look_up_similar_issues.patch* added

+1: I tried another approach to create a feature that displays issues with similar subjects. By using the existing action `"/issues/auto_complete"` in Ajax, the feature could be implemented with a little changes. The UI is as shown below (referring to the similar feature of Github).

live_look_up_similar_issues.png

#34 - 2019-10-12 09:46 - Go MAEDA

- Target version changed from *Unplanned backlogs* to *Candidate for next major release*

#35 - 2019-12-27 08:47 - Malte Blanck

Takenori TAKAKI wrote:

+1: I tried another approach to create a feature that displays issues with similar subjects. By using the existing action `"/issues/auto_complete"` in Ajax, the feature could be implemented with a little changes. The UI is as shown below (referring to the similar feature of Github).

Nice! Any chance that you release this feature as an Plugin for 4.x ?

Files

Capture.PNG	14.8 KB	2011-01-26	Andrew Betts
dups.js	2.32 KB	2011-02-07	Dustin Lambert
ro5840-integrate_dups_js_by_dustin_lambert-trunk_r7885.patch	2.9 KB	2011-11-24	Mischa The Evil
live_look_up_similar_issues.png	72.8 KB	2019-10-03	Takenori TAKAKI
live_look_up_similar_issues.patch	3.03 KB	2019-10-03	Takenori TAKAKI