

Redmine - Patch #40538

Hi, can you help me with a Version Extended?

2024-04-08 16:22 - Enzo Pellecchia

Status:	Reopened	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	Roadmap	Estimated time:	0.00 hour
Target version:			
Description			
<p>I would need to show in the roadmap, in addition to the version issues, also any linked children found in other versions of other projects.</p> <p>I tried to make an extension of the class, but then I get an error on a count method outside of this code. Probably the type of the class I return is wrong.</p> <p>Can you give me some suggestions?</p> <p>I attach the file with the class.</p> <p>Thank you.</p>			

History

#1 - 2024-04-08 19:45 - Holger Just

- Status changed from New to Closed
- Private changed from No to Yes

#2 - 2024-04-08 19:46 - Holger Just

- Status changed from Closed to Reopened
- Private changed from Yes to No

Sorry, this was closed in error...

#3 - 2024-04-08 19:56 - Holger Just

This change (i.e showing kind-of unrelated issues in a version) appears to be quite surprising. As no other views would show these issues, leading to inconsistent reports. Also, it's unclear why you are only including issues from other projects and why you are only including direct child issues from the issues assigned to the version while omitting deeper nested trees.

In any case, a better solution for your use-case is likely to have sub-issues inherit the version of their parents. This has been requested in [#6117](#) (or more generalized in [#13585](#)). Right now however, there is only a feature to set some fields of the parent issues to values derived from its children, specifically the start and due dates and the priority.

As for your specific patch, the original method returns an ActiveRecord relation while your returns an array. These behave differently which may lead to the error you have observed.

#4 - 2024-04-09 12:42 - Enzo Pellecchia

Thanks for the answer, the problem is related to the fact that the main product is composed of sub-products that are worked on by different teams with different planners. I cannot structure them as sub-projects, since other products are also needed.

So when our PM creates a main activity, he subsequently creates activities in other projects by referencing it to his main activity. Which is why he would like to see at least in the roadmap the tickets assigned to other projects that serve to close his main activity.

I tried to change in this, but not have resolved:

```
module VersionExtension
  def self.included(base)
    base.class_eval do
      def visible_fixed_issues
        # Assicurati che fixed_issues non sia nil
        fixed_issues = self.fixed_issues || Issue.none
        # Otteniamo le issue fisse per questa versione
      end
    end
  end
end
```

```
fixed_issue_ids = fixed_issues.visible.pluck(:id)
child_issue_ids = Issue.where(parent_id: fixed_issue_ids)
                      .where.not(project_id: project_id)
                      .pluck(:id)
issue_ids = (fixed_issue_ids + child_issue_ids).uniq

# Utilizziamo `where` direttamente su `Issue` per ottenere una relazione ActiveRecord
all_issues = Issue.where(id: issue_ids)

total_issues = all_issues.count
closed_issues = all_issues.where(status_id: IssueStatus.find_by(is_closed: true)).count
closed_percent = (closed_issues.to_f / total_issues * 100).round(2)

  [all_issues, closed_percent]
end
  end
end
end

# Includi il modulo nel modello Version
Version.include(VersionExtension)
```

Files

version_extension.rb	679 Bytes	2024-04-08	Enzo Pellecchia
----------------------	-----------	------------	-----------------