

Make the ID column configurable in the issues listing like the other columns

2022/05/09 18:02 - Admin Redmine

ステータス:	New	開始日:	2022/05/09
優先度:	通常	期日:	
担当者:		進捗率:	0%
カテゴリ:	Issues filter_56	予定工数:	0.00時間
対象バージョン:		作業時間:	0.00時間
Redmineorg_URL:	https://www.redmine.org/issues/23417	status_id:	1
category_id:	56	tracker_id:	3
version_id:	0	plus1:	1
issue_org_id:	23417	affected_version:	
author_id:	107353	closed_on:	
assigned_to_id:	0	affected_version_id:	
comments:	13		

説明

Currently, in the issues listing table, there are two "hard coded" columns:

first column with the checkbox

second column with the issue ID

In our custom theme, we want to have the subject in the first column in order to improve the readability of the issues and subtasks: !id_column.png!.

To obtain that, we hide both columns from the CSS, but we weren't able to display the ID column as last column in the table by default without development (or let the users to decide the ID column position).

Attached is a patch that:

- makes the ID column configurable (like the other columns)
- adds the ID column as first column in the default Redmine settings
- migrates the default columns settings in order to add the ID column as first column.
- updates the existing tests to include this change
- removes some empty spaces

journals

+1

Very useful patch!

The migration file is missing from the first patch. I've attached now the complete patch (make_id_column_configurable_v2.patch).

- "plugins/README" is deleted accidentally.
- db/migrate is too hard to test. Could you use alternative way? ----- Toshi MARUYAMA wrote: > * db/migrate is too hard to test. > Could you use alternative way?

Oh, I can see 20131004113137_support_for_multiple_commit_keywords.rb uses similar way.
source:tags/3.3.0/db/migrate/20131004113137_support_for_multiple_commit_keywords.rb

Thanks Toshi for your feedback.

I've updated the patch in order to:

- clearly apply on the current trunk
- remove the accidentally "plugins/README" deletion

For testing the "db/migrate" I used the following scenarios:

- open "settings?tab=issues" and "projects//issues?set_filter=1"
- apply the patch without migration
- refresh the admin page and observe that the default columns are still the same (without id)
- refresh the issues list and observe that the ID column is no longer displayed in the issues table
- run the migrations
- refresh the admin page and observe that the ID column now is the first column in the default columns setting
- refresh the issues list and observe that the ID column is displayed now in the issues table.

Regarding the implementation, I don't have another alternative way for the moment. The logic behind the current implementation was:

- get the current default columns setting
- check if the ID column already exists in the settings (on a standard Redmine setup (without patches and plugins) this case cannot exist, but I wanted to cover other cases where a plugin or a patch already made this column configurable)
- if exists, do nothing
- if doesn't exist, add the id column as first column
- save the new settings.

Please let me know if you still want to find another way for migrations or make other changes to the patch.

Toshi MARUYAMA wrote:

Toshi MARUYAMA wrote:

- db/migrate is too hard to test. Could you use alternative way?

Oh, I can see 20131004113137_support_for_multiple_commit_keywords.rb uses similar way.
source:tags/3.3.0/db/migrate/20131004113137_support_for_multiple_commit_keywords.rb

Yes, my implementation was inspired from that file.

Updated the patch to cleanly apply on the current trunk.

The patch is compatible with the latest trunk. @Jean-Philippe Lang, please let me know if is anything wrong with this patch.

Marius, nothing is wrong with the patch but having the issue id in first position is part of the UI. Users should not have to handle this when choosing the information that they want to see on the issue list, and I prefer not to apply this change.

Note that saved queries would also have to be fixed in a migration (id column is missing from all saved queries after applying the patch). If you want to always have it in last position, you could alter Query#columns to append the id instead of inserting it in first position.

Thanks for your feedback, Jean-Philippe.

Jean-Philippe Lang wrote:

If you want to always have it in last position, you could alter Query#columns to append the id instead of inserting it in first position.

No, I don't want to have it in last position, I prefer to let each user decide the position of the ID column in the issue list. For example, we have some projects where the ID is not added to the issue list because it is irrelevant.

CAN U CREATE PLUGIN for it or update patch plz

i dont need issue id appear in my report (pdf,cvs)

履歴

#1 - 2022/05/10 17:07 - Admin Redmine

- カテゴリ を Issues filter_56 にセット