

Redmine - Feature #42630

Add reaction feature

2025-04-25 08:49 - Katsuya HIDAKA

Status:	New	Start date:	
Priority:	Normal	Due date:	
Assignee:		% Done:	0%
Category:	UI	Estimated time:	0.00 hour
Target version:	6.1.0		
Resolution:			

Description

This patch adds a reaction feature.

- Adds reaction buttons to issues and notes, news and comments, and forum messages
- Shows a tooltip with the names of users who reacted—up to 10 names, with the remaining number shown if there are more than 10
- The feature is enabled by default and can be turned off in the administration settings

screenshot-issue.png
screenshot-tooltip-more-than-10-users.png

Please also refer to the following screenshots and GIF:

- [short-demo.gif](#)
- [screenshot-news.png](#)
- [screenshot-forum.png](#)
- [screenshot-setting.png](#)

Background / Purpose

- In my experience, there are many times on Redmine when I want to react to something without leaving a full comment. Reactions are now a common way to express appreciation, agreement, or support. I believe this feature will help encourage more interaction within Redmine.
- Many modern project management tools like GitHub and Jira already support reactions. By adding this feature to Redmine, I believe it will help modernize the platform and promote its adoption.

Details

- The icon used for reactions is thumb-up. This icon is commonly used to express appreciation, agreement, or support, and is widely adopted in tools similar to Redmine, such as GitHub, Jira, Asana, and Slack. Based on my experience, thumb-up is the most frequently used reaction icon in these tools.
- Since reactions should be widely available by nature, this patch does not add a specific permission for them. Instead, access is controlled based on the following rules:
 - Add/Remove a reaction: Any logged-in user who can view the object (e.g., object.visible?(user) returns true)
 - View reaction button: Any user who can view the object

Implementation

- To reduce memory usage and performance impact when displaying objects with many journals (such as issues), I implemented the feature to load only the minimum necessary data.
- I confirmed that all tests, including system tests, pass.

You can also check the source code on GitHub:
<https://github.com/hidakatsuya/redmine/tree/reaction-feature-dev>

Performance

The following results were measured using rack-mini-profiler and memory_profiler when opening an issue detail page. The tested issue has **50 journals, each with reactions from 10 users**. The Redmine instance is running the latest trunk on PostgreSQL in production mode.

--	--	--

	Response Time (average of 5 runs)	Memory Performance
Reactions Enabled	257ms	Total allocated: 14490218 bytes (128483 objects) Total retained: 2576895 bytes (7804 objects)
Reactions Disabled	214ms	Total allocated: 11122428 bytes (96509 objects) Total retained: 2064420 bytes (6036 objects)

Related issues:

Related to Redmine - Feature #1011: Add voting to tickets

New

2008-04-07

History

#1 - 2025-04-25 10:12 - Katsuya HIDAHA

- File Add-reaction-feature.patch added

The previously attached patches have conflicts, so I'm attaching a new patch, **Adds-reaction-feature.patch**, with the conflicts resolved. This patch includes both the implementation and test code. If there are any future conflict resolutions or updates, I will attach a new patch with the same filename.

#2 - 2025-04-25 10:55 - pasquale [:dedalus]

+1

#3 - 2025-04-25 11:08 - Kohei Nakamura

+1

#4 - 2025-04-26 07:38 - Go MAEDA

- Category set to UI
- Target version set to Candidate for next major release

I strongly support adding a reaction feature to Redmine.

It would make it easy for users to show agreement or feedback with a simple click. Since a feature like this is common nowadays, even beginners can use it intuitively. This could help activate communication by making it easier to respond.

Currently, many users post "+1" comments on www.redmine.org to show their support for the issue. If we have a reaction feature, users can react without posting "+1" comments, helping keep discussions more focused.

The proposed patch allows administrators to disable reactions if they don't need them, so it won't negatively affect users who prefer not to use it.

Many modern tools like GitHub, Jira, Asana, and GitLab already offer reaction features. Without a similar feature, Redmine might be seen as lacking something important.

#5 - 2025-04-26 08:41 - Mizuki ISHIKAWA

+1
After applying the patch, it looks really good!
The thumbs-up icon is very useful because it allows users to widely express positive reactions, such as agreement, appreciation, or confirmation, without having to leave a comment.
I would really love to have this feature available.

#6 - 2025-04-26 11:54 - Go MAEDA

- Related to Feature #1011: Add voting to tickets added

#7 - 2025-04-28 05:31 - Go MAEDA

- Target version changed from Candidate for next major release to 6.1.0

A feature like this has been requested repeatedly. You can see this from [#1011](#), which was posted 17 years ago, and its related issues.

Let's include this long-awaited feature in [6.1.0](#). I am setting the target version to 6.1.0.

#8 - 2025-04-28 18:26 - Holger Just

Thank you for this proposal!

I do have some random remarks. Sorry for the brevity.

- We may need to check user visibility in the list of users who reacted to an object and only include users visible to `User.current`. As anyone can react, this may expose non-visible users otherwise. For that, it may be possible to introduce `Reaction.visible` scope or some other way to filter users.
- I'm not a fan of returning final user names in `Redmine::Reaction::Reactable@reaction_user_names`. I think we should return actual `User` objects here rather than names (and thus rename the method). The consumer should then format the user appropriately on the view/helper or e.g. the API. This also applies to `Reaction.users_map_for_reactables`
- In the `ReactionsController`, we should first check `require_login`, followed by `check_enabled` to avoid leaking the setting value to anonymous users.
- The `ReactionsController` probably needs to check the request format and handle non-js requests. See [#39894](#) for a similar issue.
- In the `reaction_text_enabling_reactions` translation, we probably don't want to spell out the list of objects one can react to so that this list does not become outdated or incomplete if we or plugins later extend the list. This should be more generic.
- The reactions should also be included in the API responses for issues, messages, ..., possibly behind an include flag

Some suggested feature work:

- Add a column and a filter for the `IssueQuery` to filter issues which the current user (or some other user?) has reacted to. This can be similar to the watcher column / filter. * A sortable column containing the number of reactions may be helpful. * For larger projects, a column/filter with some definition of "recent" reactions might also help.
- Maybe add a set of permissions to control who can react and who can see reactions. These permissions probably need to be per object type (i.e. separate permissions for issues and forum posts). At least a "can react" permission would be helpful.
- A useful (later) extension may be to add negative reactions (or something like a set of possible emoji reactions, some of which may be positive or negative). While this may still need additional refinements, the design should probably allow this as a future extension.
- A user may (optionally?) want to be notified when someone reacts to one of their posts.

#9 - 2025-04-28 18:30 - Holger Just

One additional issue:

It must not be possible to react to objects in closed projects. This may require specific checks as the view permissions are still granted here, while changes must not be possible anymore.

Edit: This and similar issues may be solved by introducing new "View reactions" (read-only) and "React" (write) permissions, similar to the various watcher permissions. If desired, we could add a migration which sensibly grants these permissions to existing roles, e.g. mirroring the watcher permissions of the respective role.

#10 - 2025-05-01 03:53 - Katsuya HIDAHA

- File `Add-reaction-feature.patch` added

[Holger Just](#)

I appreciate your review of this patch. Please find my responses below.

In the `ReactionsController`, we should first check `require_login`, followed by `check_enabled` to avoid leaking the setting value to anonymous users.

You're absolutely right. I will fix this.

The `ReactionsController` probably needs to check the request format and handle non-js requests. See [#39894](#) for a similar issue.

I don't think the issue described in [#39894](#) would occur, as the actions in the `ReactionsController` require authentication. However, explicitly indicating that the controller expects only JS format requests improves code readability. Therefore, I will modify the controller to return a 404 response for non-JS requests.

In the `reaction_text_enabling_reactions` translation, we probably don't want to spell out the list of objects one can react to so that this list does not become outdated or incomplete if we or plugins later extend the list. This should be more generic.

I agree. The description was intended to help explain the features reactions can be used with, but it's merely supplementary and not essential. Based on your feedback, I will remove the description from the setting.

The reactions should also be included in the API responses for issues, messages, ..., possibly behind an include flag

I suggest we leave it out of the scope of this patch for now and only consider it when there is a concrete proposal or demand for API integration, including discussions around the API specification. The reasons are:

- We need to discuss both the necessity and the design of API support.
- Keeping the scope of this patch minimal will help facilitate focused discussions and reviews.

I will respond to the following points separately, but I will at least address the first one in an upcoming update:

- We may need to check user visibility in the list of users who reacted to an object and only include users visible to User.current. As anyone can react, this may expose non-visible users otherwise. For that, it may be possible to introduce Reaction.visible scope or some other way to filter users.
- I'm not a fan of returning final user names in Redmine::Reaction::Reactable@reaction_user_names. I think we should return actual User objects here rather than names (and thus rename the method). The consumer should then format the user appropriately on the view/helper or e.g. the API. This also applies to Reaction.users_map_for_reactables

I've uploaded a new patch that includes the above changes along with some additional improvements. The file name remains Add-reaction-feature.patch. If you would like to test the patch locally, please use the version with the most recent upload date. You can also view the detailed diff on GitHub here: <https://github.com/hidakatsuya/redmine/compare/master...hidakatsuya:redmine:reaction-feature-updates>

The changes included in this patch are as follows:

- Explicitly handle only JS format requests
- Check login status first to prevent configuration leakage
- Remove setting description to avoid inconsistencies if the scope of the feature changes
- Add a readonly style to reaction buttons
- Fix an incorrect expected value in the helper test
- Fix an issue where reaction button positions shifted when zooming in or out

Files

screenshot-forum.png	101 KB	2025-04-25	Katsuya HIDAHA
screenshot-issue.png	176 KB	2025-04-25	Katsuya HIDAHA
screenshot-news.png	103 KB	2025-04-25	Katsuya HIDAHA
screenshot-setting.png	166 KB	2025-04-25	Katsuya HIDAHA
screenshot-tooltip-more-than-10-users.png	96.9 KB	2025-04-25	Katsuya HIDAHA
short-demo.gif	535 KB	2025-04-25	Katsuya HIDAHA
0001-Adds-reaction-feature-to-issues-news-and-forums.patch	28.1 KB	2025-04-25	Katsuya HIDAHA
0002-Adds-tests-for-reaction-feature.patch	31.4 KB	2025-04-25	Katsuya HIDAHA
Add-reaction-feature.patch	56.1 KB	2025-04-25	Katsuya HIDAHA
Add-reaction-feature.patch	57 KB	2025-05-01	Katsuya HIDAHA