

Moved to:

<https://github.com/dear-github/dear-github>

Dear GitHub,

You have done so much to grow the open source community and make it really accessible to users. Somehow you have us chasing stars and filling up squares, improving the world's software in the process.

However, many of us are frustrated. Those of us who run some of the most popular projects on GitHub feel completely ignored by you. We've gone through the only [support channel](#) that you have given us either to receive an empty response or even no response at all. We have no visibility into what has happened with our requests, or whether GitHub is working on them. Since our own work is usually done in the open and everyone has input into the process, it seems strange for us to be in the dark about one of our most important project dependencies.

The problems we most frequently have, and our best ideas for how to address them, are:

- Issues are often filed missing crucial information like reproduction steps or version tested. We'd like issues to gain custom fields, along with a mechanism (such as a mandatory issue template, perhaps powered by a `newissue.md` in root as a likely-simple solution) for ensuring they are filled out in every issue.
- Issues often accumulate content-less "+1" comments which serve only to spam the maintainers and any others subscribed to the issue. These +1s serve a valuable function in letting maintainers know how widespread an issue is, but their drawbacks are too great. We'd like issues to gain a first-class voting system, and for content-less comments like "+1" or ":+1:" or "me too" to trigger a warning and instructions on how to use the voting mechanism.
- Issues and pull requests are often created without any adherence to the `CONTRIBUTING.md` contribution guidelines, due to the inconspicuous nature of the "guidelines for contributing" link when creating an issue and the fact that it often contains a lot of information that isn't relevant to opening issues (such as information about hacking on the project). Maintainers should be able to configure a file in the repo (interpreted as GFM) to be displayed at the top of the new issue / PR page instead of that link. Maintainers can choose to inline content there and / or link to other pages as appropriate.

Hopefully none of these are a surprise to you as we've told you them before. We've waited years now for progress on any of them. If GitHub were open source itself, we would be implementing these things ourselves as a community—we're very good at that!

Signed,

Adam Bradley (@adamdbradley), maintainer of [Ionic](#)  
Addy Osmani (@addyosmani), maintainer of [TodoMVC](#)  
Andreas Tolfsen (@tolfsen), maintainer of [Selenium](#)  
Ariya Hidayat (@ariyahidayat), maintainer of [PhantomJS](#)  
Cătălin Mariș (@alrra), maintainer of [HTML5 Boilerplate](#)  
Christopher Chedeau (@vjeux), maintainer of [React Native](#)  
Dave Methvin (@davemethvin), maintainer of [jQuery](#)  
Domenic Denicola (@domenic), maintainer of [WHATWG Standards](#)  
Forbes Lindesay (@ForbesLindesay) maintainer of [Pug \(formally known as Jade\)](#)  
James Kyle (@thejameskyle), maintainer of [Babel](#)  
Henry Zhu (@hzoo), maintainer of [JSCS](#)  
John-David Dalton (@jddalton), maintainer of [Lodash](#)  
Juriy Zaytsev (@kangax), maintainer of [Fabric.js](#)  
Ken Wheeler (@ken\_wheeler), maintainer of [Slick](#)  
Nicholas C. Zakas (@nzakas), maintainer of [ESLint](#)  
Pascal Hartig (@passy), maintainer of [TodoMVC](#)  
Sam Saccone (@samcccone), maintainer of [Marionette](#)  
Sindre Sorhus (@sindresorhus), maintainer of [Yeoman](#)  
Ashley Williams (@ag\_dubs), Developer Community Manager, [npm](#)  
Trek Glowacki (@trek), a maintainer of [Ember.js](#)

---

## Scratchwork appendix:

### Frustrations:

- Issue management and notifications do not scale for larger projects, they are overwhelming and it actively contributes to burnout of open source maintainers.
- No validation that any part of CONTRIBUTING.md has been adhered to before an issue is created
- Auto subscribing users when they get mentioned. This encourages random users to mention maintainers in issues wanting direct support leading to a lot of email noise.
- The existing pull request workflow and green “Merge” button encourage merge commits, which many larger projects do not allow.
- Excessive comment noise due to +1s and other nonsense interactions
- People adding “famous” people as contributor for fun to their projects, creating frustration for them and spam in the activity stream.

### Improvements:

- Allow custom fields in issues (e.g. library version, language version, operating system)
  - Make fields required (e.g. library version or link to a test case; making such information mandatory would dramatically reduce noise)

- Show contributing guidelines before opening an issue (yet another noise reduction)
- Show count of how many people will be notified next to the issue creation button
- View notifications without marking them as read
  - Allow disabling automatically marking them as read, so they have to be manually marked as read.
- Flag notifications or issues for future follow up (i.e. create a personal/private flag)
- Turn unintuitive search query system into GUI
- Auto-search for "similar issues" based on the words in the issue title and body when the issue is created, to prevent dups - similar to how Stack Overflow already works
- Select multiple issues and then close with a single comment
- Ability to merge issues
- Ability to move issues to other repos
- Explicitly eliminate content-free replies like "+1", or convert them into a rating system for issues so watchers aren't constantly notified! A separate rating system would be nice too.
- Voting system for proposals
  - Something that surfaces votes at an issue-viewing level that can be sorted easily by counts/labels. This allows distilling the highest priority bits to work on.
- Add support for custom reply messages (e.g. "Can't reproduce...", "Please attach a test case...", "Please use forum for questions...", etc.)
- Add support for default placeholder message when opening an issue (similar to what Google Code did — "What steps would reproduce a problem? ... Expected result: ... Actual result: ... etc.")
- Better tools for addressing tracker spam comments/issues
- Don't make it so easy to submit bad PRs → For example, remove the default commit title when creating a commit from the GitHub UI. When I see "Update readme.md", I know I'm in for annoyance.
- Ability to block users from an organization.
- Allow assigning issues to anyone who has contributed to the issue, not just project maintainers. This way issues can be assigned back to the OP when more info is needed.
- Support explicitly marking an issue/pull request as "Needs Revision"/"Needs More Info" and then have the tag automatically removed once someone comments on the issue/updates the code in the pull request.
- Support "private labels" for issues. Users should be able to add private labels to any issue, even if it's a project they don't maintain. They should then be able to use them for querying/filtering.
- Allow automation rules, such as automatically notifying or closing when issues have been untouched for a long time.
- Support linear histories in the big green merge button. The project maintainer should be able to specify whether merge commits should be created, or the change should be rebased on top of master.

- Support the ability to change the target branch in a PR without having to close and reopen a new one -- This way you do not have to lose the comments and discussion that may have happened in the original PR
- Allow more than one format for reviewing issues. For example, displaying all issues within a spreadsheet format, sortable by columns.