

**Issue Tracker Overview & Research**  
**Github Issue Tracker Redesign**  
**By: Javan Friedel**

**For:**  
**RouteThis Application**

## **Problem Overview: Github Issue Tracking Redesign**

### **Deliverables:**

- 1) **Designs** for an issue tracker that have mockups for the following:
  - a) A list of existing issues with the ability to add a new one
  - b) The form/process a user would fill out to file a new issue
  - c) A view that shows the details of a logged issue
- 2) Some kind of **functional set of components or in-browser demonstration** of some of the designs
  - The implementation doesn't have to actually WORK (i.e. no databases or backend)
  - It should demonstrate the look-and-feel of the components and pages

## Research:

### **Target Audience:**

Tech savvy individuals, large companies and coding groups with a large collaborative focus.

### **First Glance: Function / Clarity**

- The system is built more for developers than for users.
  - This makes sense from the target market of larger groups on “the in” of these projects. However, for it being heavily about large projects the scalability of the issue tracker seems dismal.

### **List of Issues**

- No clear way of sorting by popularity or severity. Default is in order of which ones are posted. This makes sense as a contributor, but might not make sense as someone there to post an issue.
- A decent amount of projects use the issue tab for feature requests as well. Might be useful to split this functionality entirely. This could also allow authors to disable the tab, if it is not a project that supports feature requests. Could also have separate systems, for instance "liking" a feature request could show popularity / need for it. If deemed the issues tab does not need this functionality.
- UI uses mostly single line information for the title and tags and does not line up to sort options above. This causes a disconnect visually from what you have sorted to why it is being posted. For instance, if I sort by how many reaction:dislikes something has it just serves me the list without displaying how disliked it is. The first entry could be 1000, the 2nd, 3 and I would never know from the list view. Suggestion: Dynamic view based on what category you sorted by, showing you the details of that sort. This does not cause extra space to be used and only adds clarity.
- Ease of sort for other categories is hard as well, when searching by assignee it just gives you an alphabetical list, which is pretty arbitrary when dealing with huge projects. This list should also tie into a score list of contributions to easily see who is most likely to have an issue assigned to them. Additionally this would be easier for people to see their general workload in issues. This similar problem occurs in the label search category, it is sorted alphabetically and not by most used or important tags. Making it harder to sort by tags in general.
- Overall suggestion to make the sort better; in the categories show index information about how many times that has been listed, and rank them by that category. For instance, if someone has 10 issues assigned to them, and the next most is 5, they should appear in that order with that information displayed beside their name. Same with the labels etc. If no index information is available as in the case when it does not occur, default to alphabetical from that point on.
- Filters are very good, as this completely depletes the list into specific categories. Functionality-wise I believe this should be left unchanged, however, visually it could use some work. The commands it uses in the search bar to make it work could be done with tab boxes instead with an x on them to the right, allowing you to take out parts of the search default parameters with a simple click. Additionally to this, the popup to clear what you have searched shows up outside the box and below it, slightly disconnected from the search bar itself. I would like to see an inset clear tab on the right side of the search bar that does the same functionality and does not change the layout of the web page itself.

- The “New issue” button brings you to a landing page that basically displays what template you would like inserted into your issue, as well as guides you to the policy around security issues if that applies. If the feature requests are not separated as previously mentioned or templates need to exist at all, I would like to see an optional drop down version of new issue ticket to immediately select your template instead of going to a unrequired landing page. Speeding up the process, reducing load times, and burden on hosting.
- If sort changes work well enough I do not believe the secondary “labels” and “milestones” tab near the New Issue Button is necessary, as their functionality will mostly be integrated into list. Consider either removing entirely, and expanding the search bar, or swapping it with the Open / Closed Functionality. (Most Likely the latter, as this will visually differ them more as that is also a slight clarity issue as it stands.
- “Pinned Issues” Has a lot of potential, this box layout adds a clean highlight of issues. This could also be used for most recent or most popular by default if not other pinned issues exist. Could also utilize a combo, like one pinned issue(slight graphical change) and then rest default to most recent or most popular.
- Icons should be made to go beside the sort items like the issues and milestone tabs have above the list view.

### **Form / Process for New Issue**

- “Labels” can only be applied by contributors with write access. And can not be assigned by the writers of an issue. (public facing might have a reason to not include, private facing label suggestions seem to be a must though).
- Not well connected. Can not point to specific code file, or version of project, despite these things already well linked within github. And already heavily requested by dev teams in the actual request. If properly linked, developers could sort issues by version / file in the program. This could also lend to being able to pull reports about a development without the need to manually go down every new issue thread. Could also set up alerts when an influx of problems start to occur with a specific file. Allowing you to connect seemingly unconnected issues easily.
- Utilizes the same 1200px width wrapper as the rest of the site, however, for the new issue pages it reserves the last 200px ish column for tips, and is left mostly blank otherwise. This area could be used to display the most popular issues, as they might realize their issue is similar to another one. Additionally this list could update based on its similarity to what you are filling out. This would reduce the amount of duplicate issues being posted. Otherwise, it could also be used for some elements that aren't very prominent as it is. For instance, who to mention in your post or to reference a different issue or pull request.
- The entry box supports markdown, which is awesome, but the preview tab is exactly that... another tab. Would be nice to see you be able to live preview what you are typing. Either in the same window as what you are typing or in a vertical split view. That or you make the preview a button that transforms what you typed into a preview kinda like an on/off switch. If you attempt to click in the field again it could auto default back to normal typing mode, reducing the amount of clicks on the user to utilize the preview function.
- As mentioned earlier, Labels should be able to be applied by the people posting the issue, I suggest this gets added as a button within the title tag, and should simply add the tabs to the end of the title you are typing as this is how they will display on the list as well.
- The “hint” that markdown is supported is at the bottom of the page which doesn't make sense in priority to information importance. As I plan on removing the “preview” tab, this could be added to the top beside the editing quick options. As both the preview tab and the write tab could be removed as where to write is pretty obvious by itself.
- The markdown link does not look clickable, underlining “markdown” might make people think they can click it for more information.

### **View that Shows Details**

- Comment Chain is linear, you must add to the previous comment. As in the words of github, you cannot “fork” replies in the chain.
  - This makes sense for productive group discussion within smaller teams, but large projects could become clustered very quickly.
  - Suggestion: A chain conversation system showing only the first comment to the reply with an expandable menu to see farther down chains.
  - By default the most “liked” or “commented” answer will have their chain auto expanded as the first comment.
  - As this is a big change in hierarchy I would also advise a “timeline” tab that works chronologically as it does now.
    - As this will split the conversation I will have to sketch / mockup ways to rectify this.
- Shows list of Participants in the conversation, but has no indication to who those people are other than profile pictures. Might be useful to “star” or highlight the contributor on this specific repository. Maybe even have a lvl system based on scoring how many comments, commits etc they have completed. Showing their overall involvement in the project. (this functionality already exists for projects in ranking participation, but is not displayed other than in the contributors tab.
- Ability to subscribe to alerts when thread is closed, but no quick way to jump to the entry when it was closed. (However if fixed in a pull request it will link the entry.) For usability this should be added to the top near closed. This will also ID the closed tag and make it easy to link directly to it externally. (Which can already be done with comments)
- The new issue button is in-line with the title, which makes it float there a bit, and cause the title to go to a new line sooner. I would much rather see it centred to the sub-title which tends to be much shorter, giving more space for title and in line more with the natural view of the dividing line.
- Dividing line on the comment is not necessary, rather it should visually display the chain if chronological is solely kept.
- Order of sub-menu seems out of place, notifications should be top, followed by labels as these are functional or category statements and help define what is being looked at.
- Same issues of editing text as the submission form, make sure it stays consistent with this input box.
- On larger projects many comments definitely become unorganized, however comments on a single issue never really reach insane numbers. Chronological could work, but on anything over 50 comments I think some sort of navigation or linking could become

useful. A back to top button after scrolling far might also be useful as they do not have a fixed navigation