

Liferay Community

Liferay Community Verifier Program Guide

January 12, 2011

Welcome to the Liferay Community Verifier Program! This document is intended to be used by members of the Liferay Community Verifier Program. The intention of this effort is to help the Liferay community in organizing and cleaning up the set of Liferay Issues filed at issues.liferay.com.

If you have questions, comments, or suggestions about what you are supposed to do or what you are experiencing, just email the community manager (James Falkner - james.falkner@liferay.com) or program manager (Cynthia Wilburn - cynthia.wilburn@liferay.com). You can also find many Liferay developers and Verifier members on **IRC** (a public chat room). Details for connecting to IRC can be found on the [Participate page](http://www.liferay.com/community/welcome/participate) at <http://www.liferay.com/community/welcome/participate>.

Additional JIRA Privileges

As a member of the Community Verifier Program, you have been placed into a special group called “liferay-community-verifiers” and given special privileges at issues.liferay.com. These additional privileges include:

- Ability to Edit Issues
- Ability to Assign Issues to Liferay Staff
- Ability to Edit Own Comments
- Ability to Delete Own Attachments
- Ability to Set Issue Security

You will use the additional privileges in your work as part of this program.

Timeline

The Community Verifier Program is an ongoing effort across multiple Liferay releases. Many issues are filed on various releases, most of which are valid at the time they are filed, but due to the large number of community members working on the code, the structure of the project, and the passage of time, many issues are resolved without ever knowing an issue was filed. This program is therefore always in effect, and the membership and level of activity per member will vary over time. In addition, new community members arrive and old community members depart, so there will always be “knowledge leaks” that can be filled by experienced community members such as yourself.

Notes

Support QA User

This is a special user within JIRA that is a bucket from which Liferay QA works from. In many scenarios below, you will be assigning bugs to this special user. It can be found in the same drop-down pick list that all users are found. When you wish to assign a bug to this user, click the “Edit” button to edit the bug’s metadata, and select the **Support QA** user in the “Assignee” field. It’ll look like:



Liferay 6.1 and Licenses

In the below scenarios, when reproducing bugs that have enough information to do so, for now you will be reproducing the bug on the *latest Liferay 6.1 CE release* (currently 6.1.0, also known as “GA1”, which was released in January 2012).

Once the EE version of 6.1 is released, the Community Verifier team will begin using that release instead. This is because Liferay does not issue patches or service packs for the CE release, so the knowledge that a bug still exists in a CE release, while useful to know, is not important from Liferay QA’s perspective, because Liferay will not issue a patch to fix the issue for CE. Of course, as a result of your triaging of this bug, if it does indeed still exist in the latest *EE* release, then it is *guaranteed* to exist in the latest CE release also (barring an unfortunate regression in an EE release), and will be addressed as resources allow. Therefore, it is more useful for the Community Verifier team to verify against EE rather than CE, *once it is released*.

One may argue, then, that we should instead verify against *trunk* (the absolute latest, bleeding edge development snapshot). There are some merits to that, however there are a couple of reasons against it:

1. Trunk is unstable. There are many changes and moving parts that occur on an *hourly* basis during active development. You may not even have a functioning Liferay. And there may be many false negatives - the bug may appear to be fixed, but that’s because of another bug preventing the original bug effects from manifesting themselves.
2. Most fixes on trunk are ported to EE releases. There is a time lag of course, but for most bugs (especially those older bugs which we are dealing with in the Community Verifier Program), they have either been ported to EE, or the bug still exists in EE. Either way, testing against EE will either find it, or not, and this is a definitive statement about whether we should fix the bug.

What is Expected of Community Verifiers and You

The following set of guidelines should help you get started right away. The basic outline is to select one of the many open bugs that need attention, and follow the below steps to triage the issue. Some of these issues are still present in the latest Liferay code, while others have been fixed or are no longer reproducible, and need to be closed.


1. **Choose a ticket to work on.** You can use the [“Community Verifier: Bugs needing Verification” dashboard](#) to quickly see which bugs need attention, organized by functional component. These bugs have an Issue Type of “Bug” or “Regression Bug” where there is no fixed-in version set and the reporter is not a member of the group “Liferay-staff”. Any bug from this list is a candidate bug. You may alter this filter to only show bugs in specific components that you are more comfortable or familiar with. There are currently ~400 issues on this list. If each member of this team triages a couple of bugs a week, after a handful of weeks we can completely eliminate this list, and that would indeed be a very momentous occasion and one that you could all be proud of. This is the goal!

2. **Add a label:** When you choose a ticket, the first thing you

Labels:

community-verifier

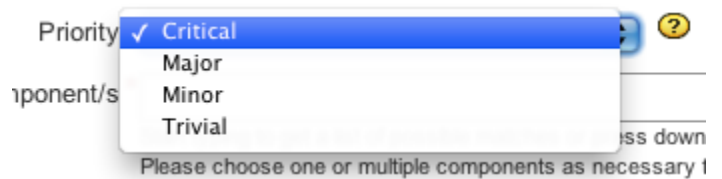
should do is add the following label: **community-verifier**.

This label will ensure that two people don't choose to work on the same bug (the above JIRA filter eliminates these from the list) and we can produce nice looking graphs tracking this group's activities. You can do this by clicking the small pencil icon next to "Label" field near the top, and adding the label (it will auto-complete for you). Be sure not to delete any existing labels! You can also optionally 'watch' the ticket, so that you receive email updates when changes are made. Click the  **Watch (3)** icon to enable this.

3. Is it a bug: Next, you must quickly assess whether the ticket contains information describing an true bug OR if the reporter was looking for help by asking a question in the bug report and requires support from the community. If it is the latter, please do the following:

- Assign the ticket to **Support QA**
- Enter the canned response: **<Reporter_Name, Thank you for your report. Have you had the opportunity to discuss this issue on the community forums? If so, were there other users who were also experiencing this problem? Thanks! This issue will be closed.>**

4. Adjust the priority if needed. Check to see if the severity field is accurately set to reflect the problem the user is experiencing. If it is not, then change it to the appropriate severity level (After reading and understanding the definitions of severity levels on the [JIRA wiki](#)).



Any changes to this field should reference the JIRA Wiki and include a response such as this: **<I am changing the priority on this ticket to Minor/Major/Trivial based on the information found in our [JIRA reporting wiki](#) regarding the criteria for priority/severity fields. if you feel this was done in error, please provide additional information so we can re-evaluate the issue.>**

5. Is the issue reported against Liferay 6.1? If not, skip to step 6. If so, any ticket that is reported against the Liferay 6.1 CE release family (6.1.x) should be tested using the latest 6.1 CE release (currently 6.1.0) to determine if it is still reproducible. Tickets that you are working on (attempting to reproduce on 6.1.0) should have a comment that both informs the reporter and the other team members that you are working on it. **<Thank you for the report. As part of our community verifier program, I am taking a look at this issue. If it is no longer a concern or you have additional information that will help in troubleshooting it, please let me know. >** This might prompt the original submitter to supply more information, or (better) tell you that the bug has been fixed or is no longer a bug, in which case you can cease work on it, assign it to **Support QA**, and add a comment such as "This issue is no longer a bug, verified by submitter. This issue can be closed".

- When you cannot reproduce the ticket, please add the following comment **<Reporter_Name, thanks for the report. I tried to reproduce this issue on Liferay Portal 6.1 CE but could not. Here are the steps I tried (list steps). Please ensure you have a fresh environment with no other issues that may interfere with the bug. If you still experience this issue, please let me know how to reproduce the issue and we can re-visit it. If I cannot reproduce this issue, it cannot be fixed. Thanks!>**. At this point, we want to give the reporter time to respond. If there are other participants in the comment threads or links to the forums, please take a moment to see if there is useful information there to determine if the issue is resolved.
- If the reporter does not respond in **10 days** or agrees that the issue is resolved, please assign the ticket to **Support QA** and recommend closure via a comment such as "This issue has been resolved and can be closed" or "This issue is not reproducible and the submitter has not responded within 10 days. This issue can be closed". JIRA allows you to create your own filters based on last commented dates.

- If you believe the ticket is a duplicate of another, please add the link to the duplicate issue by linking via the “relation” feature of JIRA (or simply mention it in the comments), assign the ticket to **Support QA** and recommend closure as duplicate. Note that when viewing an issue, you can see a list of “related” issues (possibly pointing you directly to a duplicate bug) by clicking the “Similar Issues” toggle: Similar Issues: [Show 5 results](#) >
- If you can reproduce the bug on the latest Liferay 6.1 CE release, please make sure that:
 - i. The steps to reproduce are clear in the ticket
 - ii. The severity field is set properly (see above)
 - iii. The environment field is accurate (especially when you know the bug is related to the environment, e.g. a SQL Server bug or a GlassFish issue)
 - iv. The Liferay Component(s) in which the bug occurs is properly identified (click the “Edit” button to alter this field and add any necessary component tags):

- v. The affected version includes **6.1.0 GA1**. Since you verified the bug you should indicate this via the “Affects Version” field:

- vi. Then assign the issue to **Support QA** and they will move it to the appropriate backlog or sprint. Liferay QA has filters set up against Critical tickets as well as those with the component set for “Security” but if there is a particularly urgent issue, please feel free to email Cynthia directly at cynthia.wilburn@liferay.com to get it escalated.

- vii. Don’t forget to click “Save”!

- Note: Environmental issues have separate components (Database, Application Server, Javascript, Browser Support or Alloy UI). We have specialized developers that handle these issues as they require a different skill set to resolve.

6. Any ticket that is reported against a previous major release branch (6.0 and 5.x) and below must be verified in 6.1.0 CE GA1 in order to remain open. If you feel it is easy to attempt to reproduce this against Liferay 6.1, then you may do so (and follow the steps above, as though the ticket was filed against 6.x), however we have historically requested first that the original submitter to download the latest CE and verify if the issue still exists before spending time verifying these tickets because so many of them are duplicates at this point. The following canned text reflects our stance on this practice: **<Have you had the opportunity to download the latest 6.1 CE and verify if this is still an ongoing issue? There are hundreds of fixes between CE releases but I can update the affected version if you are able to verify its status.. You can get the latest version of Liferay Portal at <http://www.liferay.com/downloads/liferay-portal/available-releases>. If the issue cannot be reproduced in the latest 6.0 release, the issue will be closed in order to keep JIRA up-to-date.>**

A special word from Cynthia

This task that you are taking on provides a great service to the Liferay Community and I want to personally

thank you for being willing to step in and help. Over the past two years, I've worked to clean up and organize JIRA so that it is a reliable source of information as well as the system by which we implement empirical process control. At times, it has felt like an impossible mountain to climb as tickets came in faster than we could sort and verify. So again, thank you. This work is important.

I've learned by experience that closing JIRA tickets can create some issues with the community. It can be difficult to determine when to consider an issue worthy of closing. Did the reporter just not have time to get back to me? Is there some odd environmental thing going on or the issue is so intermittent that we can't track it down? Even when we know an issue is resolved, our CE users often have to wait to get the .next CE in order to benefit from it and if they can't patch it, they want the ticket left open. As a former QA person, I can tell you that I believe every bug report is important and must be looked into to the best of our abilities. In the end, we can reopen any ticket that is closed in error so this is something to keep in mind should anyone get a little upset by some of the formal ways in which we resolve tickets.

Thanks,
Cynthia Wilburn
Senior Program Manager
Liferay, Inc.