Reporting Bugs

Is it really a bug?

We appreciate your efforts to report a potential bug with OpenMRS. Before you do, please ask yourself the following:

- Am I running the latest version of OpenMRS?
  - If using an older version, is it still supported?
  - Check the Unsupported Releases (EOL) page to find out.
- It is OK to report a bug in an older unsupported version, but it will not be fixed in that version. You will need to upgrade.
- Has the bug been reported already?
  - Search the various OpenMRS sites to see if anything similar has been reported or discussed in the past. You should also search for any specific error messages you are seeing, if applicable.

Reporting Security Issues

If the issue that you want to report is a security exploit, you should not publicly disclose the details, since that would tell attackers how to exploit OpenMRS production servers in the field. Report any security vulnerabilities to security@openmrs.org (a private list of long-time OpenMRS developers) giving details about the exploit, and someone will follow up via email.

Temporary workaround

Due to a temporary bug, if you send this email from an account that matches your account on OpenMRS Talk (if you have one), it may be rejected. If so, please click here to send a private message to the "@Security" group via OpenMRS Talk.

To send a private message on Talk, you must be Trust Level 1 (basic user) or higher. If you are a new user (Trust Level 0), you may receive a message stating that you cannot send a private message on Talk. You can find out your trust level on OpenMRS Talk by logging in, click on your user profile in the top right & clicking your username to reach your user page, then the Expand button at the top right corner of the banner. Below the expanded banner, you should see several details (e.g., when you joined, etc.) including your Trust Level. If your Trust Level is 0, then this workaround will not work for you. You can achieve Trust Level 1 by continuing to read posts and browse topics on Talk or you can contact the help desk for assistance.

File a bug report

If you know the problem is related to the function or performance of a specific add-on module you are running, please consult the Module Repository and that module's documentation about how to get support. If you believe the problem is with OpenMRS itself, or if you’re honestly not sure, you can file the bug report in the OpenMRS Trunk project in JIRA. You'll need to have an OpenMRS ID to do so.

Regardless of whether you're reporting the bug to the OpenMRS core team or to a module developer, to help those developers understand the problem, please include as much information as possible:

1. A clear description of how to recreate the error, including any error messages shown.
2. The version of OpenMRS you are using.
3. The type and version of the database you are using, if known.
4. If you are using any additional modules, custom templates, stylesheets, or messages, please list them.
5. If applicable, please copy and paste the full Java stack trace.
6. If you have already communicated with a developer about this issue, please provide their name.

We encourage you to use the method above to report bugs. However, if you are not comfortable creating a traditional bug report or are otherwise unable to do so, you may use our simple bug report form.

What happens next?

For OpenMRS bugs, our core development team will review your new bug report, attempt to verify it, and prioritize it. Once verified, they will be classified as "ready for work" at which point a developer can start working to resolve the issue. As the bug reporter, you will receive e-mail notifications from JIRA (unless you opt-out) with updates.

Once a developer has fixed the bug, it will be reviewed, tested, and included in an upcoming release of OpenMRS. You can follow the progress of the issue in the tracking system to determine what release fixes the bug.

How is priority determined?

We use the following definitions when prioritizing tickets:
1. **Blocker**

   Blockers are issues that must be addressed before the milestone (or possibly development) can proceed. Blockers may delay all other issues, and it is worth assigning multiple developers to a blocker issue to get it resolved as quickly as possible.

2. **Must**

   Critical issues *must* be addressed for their assigned milestone. Critical issues are those that have been attached to (and possibly help define) a particular milestone and the milestone should not proceed until they are addressed. If critical issues are falling behind, then we will direct resources to them as needed. Unresolved critical issues may delay a milestone.

3. **Should**

   Major issues *should* be addressed for their assigned milestone; however, if resources are not available, then the issue may get moved to another milestone instead of delaying the assigned milestone. Unresolved major issues typically will not delay a milestone.

4. **Could**

   Minor issues *could* be addressed for their assigned milestone, but only if resources permit. For major and minor releases of OpenMRS, we try to ensure that at least ten minor issues are being addressed if not many more. Unresolved minor issues will not delay a milestone.

5. **Non-essential**

   Trivial issues are generally only assigned to a milestone if someone has volunteered the resources to move them along. Unresolved trivial issues will not delay a milestone.

6. **TBD**

   To Be Determined issues have not yet been prioritized. This is the default prioritization for new issues until/unless the issue has been prioritized.