



Get bugs fixed faster with the right description

Nobody likes bugs. But when they occur — get them fixed fast by following these guidelines to deliver all the information your developers truly need.

Use an issue tracker

You don't want your bug to be forgotten about and you want it to be top priority, so make sure you use an issue tracker (it doesn't matter which one). This will ensure your team knows its priorities and nothing will get lost.

Include reproduction steps

Don't write "the save button is broken". In most cases, your team won't know what that means. Even though it sounds dumb at first, not everything is as obvious as it seems to you. Bugs may be encountered only when entering specific data, taking a specific route to a page or any other steps. It may matter who you're logged in as or what button you click. If you supply your team with that information upfront, you won't have delays later.

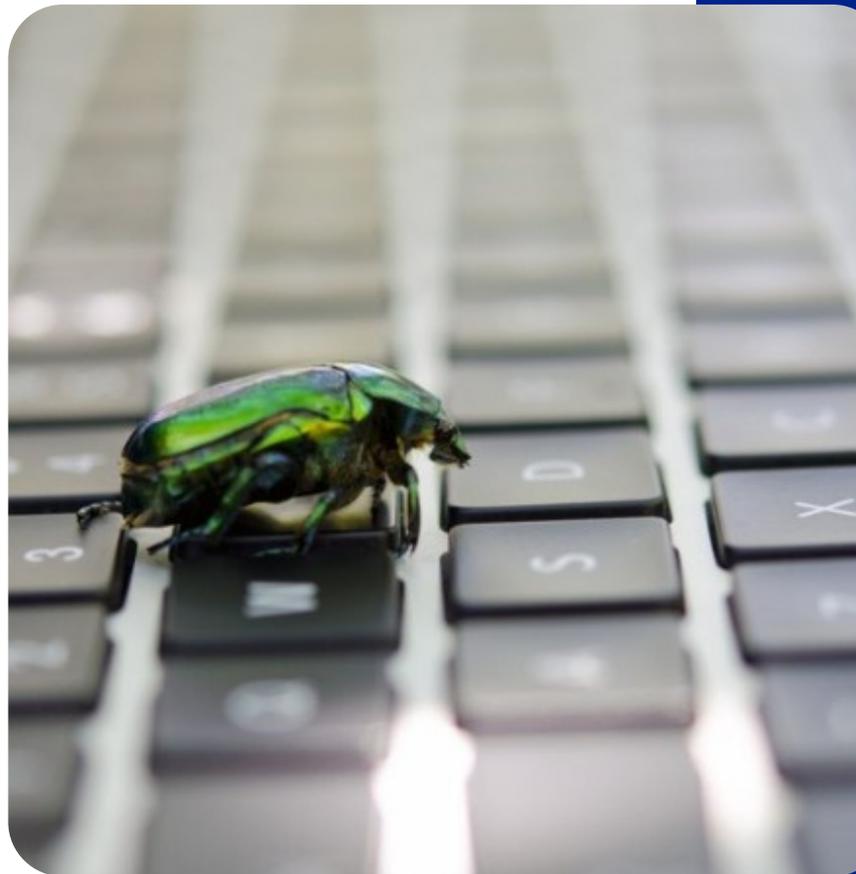
Make sure your issue contains concrete steps to get to the problem (including direct URLs, user logins, button clicks, ...) in an ordered list, e.g.:

1. Go to the page with URL X
2. Log in as user Y
3. Click button Z

Expected vs current result

Let's take another look at the commonly used bug report "the save button is broken". This literally includes no information about what happens and what is supposed to happen instead.

Imagine you or somebody else going over the bugs in a month's time. Do you remember exactly what didn't work? Probably not. Will you recognise it when that problem happens again? Maybe. But you'll have no trail of what was done, what was fixed and why things



went wrong. Thus, right below the reproduction steps, write down the expected result - given a user performs the steps above, this is what the result should be. Below, state the current result. Describe what currently happens instead of the expected result.

For example:

Expected Result: The edit page opens in an overlay.

Current Result: The edit page opens in a new tab.

That way, everybody knows exactly what's wrong now, how to fix it and you can come back to it if it comes up again in the future.

Make screenshots

A short but good one: make a screenshot of the current result and attach it to the issue. If you still think you need to open Paint to do that, here's a list of free tools which will create screenshots for you very quickly.

If you can, highlight the areas you're talking about with red arrows or circles. Don't ever write on the screenshots as the text cannot be copied. Use the issue itself to describe the problem.



Add meta information

For even more clarity, state the version of the app which you're using, the environment you were on (dev / stage / production) and your browser including its version number. Again, this will help everybody to find and fix the problem as soon as possible.

Prioritize correctly

If you want to keep a healthy relationship with your team, don't set every bug as "high priority". If you assign everything as high priority, not only will your team stop taking your bugs seriously at some point, the really important bugs will actually drown in the list of other "high priority" tasks and won't get done fast enough.

So, to get the bad boys fixed quickly, select from the following priority levels:



Blocker: Users cannot accomplish their main goal, and there's no (obvious) workaround for them (e.g. not being able to find friends on Facebook).

Critical: Users can only accomplish their main goal by using some sort of workaround (e.g. not being able to find a Facebook friend in their friends list, but being able to search for them).

Major: A significant amount of users feel uncomfortable accomplishing their main goal (e.g. the friends search on Facebook is suddenly terribly slow, but works).

Minor: Users can't accomplish their side goals (e.g. the poke button has disappeared on their Facebook profile).

Trivial: Cosmetic issues (e.g. the poke button on Facebook uses the wrong font).

Recap: Get bugs fixed faster by

- Using an issue tracker
- Including reproduction steps
- Naming the expected & current result
- Making screenshots
- Adding meta information
- Prioritizing correctly

.....and by the way, you can actually try avoiding bugs upfront. Decrease the number of bugs and don't waste your precious time and effort on fixing them! Wanna know how?

Visit the blog section at www.trustshoring.com and find much more helpful information and tips from Trustshoring. Here is what our clients say about our team:



"Absolutely delighted to work with Victor & his team. A really friendly & professional step by step process to help us find great talent to work with."

Shikhil Sharma
Founder at ASTRA