

Definition of done

2022-01-16 15:48

Contents

1	Definition of done	1
----------	---------------------------	----------

1 Definition of done

This definition is not meant to be petty bureaucracy. It's meant to be the grease that allows the gears of project to turn smoothly with minimal friction. The goal here is to enable smooth, speedy improvement without having to frequently go back to fix things.

When the software, automated tests, documentation, or web site produced by the project are changed, the change overall should make things better in some way: a bug is fixed, a feature is added, the software becomes nicer to use, the documentation more effectively communicates how to use the software, the tests cover more of the functionality, the code is nicer to maintain, or something like that.

At the same time, the change shouldn't make things significantly worse in any way. A change that, say, makes the software ten times as fast, but adds a ten percent chance of deleting the user's data would not be acceptable.

For changes to this project to be considered done, the following must all be true:

1. New functionality and bug fixes are verified by automated tests run by the `./check` script.
 - if this is not feasible for some reason, that reason is documented in commit messages, and an issue is opened so that the tests can be added later
2. The build and tests run by GitLab CI finish successfully.
3. There has been sufficient time to review the change and for interested parties to have tried it out.
 - the time needed depends on the scope and complexity of the change
 - a quick, easy change can be merged at once
 - a complex change should be open for review and testing for a few days

If all of the above conditions are met, the change can be merged into the main line of development by any person authorized to merge on GitLab. The merge will eventually, automatically trigger a build of Debian packages by Lars's personal CI.