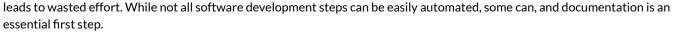TechWell Insights

From Documentation to Automation

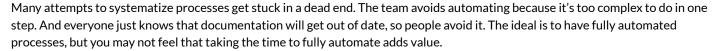By Steve Berczuk (/users/steve-berczuk) - January 13, 2020

The Agile Manifesto (https://agilemanifesto.org/) emphasizes working software over comprehensive documentation. That's not a statement against documentation, but rather an acknowledgement that only some documentation is useful.

I'm not a big fan of extensive documentation, especially around development processes. But in the absence of some sort of record, it's hard to maintain a repeatable process, and even harder for new people to contribute efficiently.

A defined, repeatable process frees people from having to spend energy thinking about solved problems, and an automated one makes this even easier. Even if this isn't immediately possible, you can move toward creating executable documentation, which can improve efficiency and repeatability.

Written processes have high overhead and are hard to maintain. Documentation takes time to write and more time to keep current, and when it's not up to date, it's confusing and leads to wasted effort. While not all software development steps can be easily automated, some can, and documentation is an essential first step.

Many attempts to systematize processes get stuck in a dead end. The team avoids automating because it's too complex to do in one step. And everyone just knows that documentation will get out of date, so people avoid it. The ideal is to have fully automated processes, but you may not feel that taking the time to fully automate adds value.

An agile mindset can help here.

To automate executable documentation, you need to know what it is you are trying to do and establish a common understanding of the steps. The second time you find yourself repeating a task that is more than a couple of steps, create a wiki page specifying what you do with enough detail that someone who knows your environment can complete the tasks. You now have a documented repeatable process.

As people use the documentation, extend it to include executable steps. So rather than simply stating, "Stop the process," you might add, "ssh user@server (mailto:user@server) sudo process stop." You are now on your way to an executable document. These code snippets can then become the basis of scripts for partial automation.

Your goal is to figure out how to automate the remaining steps. The documentation can be to run those steps, or even to point you to a services page that allows you to execute a process with a click.

There are some principles to keep in mind when following this path:

- Favor "help" information in a script or application over external documentation, as this will reduce the likelihood of docs getting out of sync with the actual process
- Maintain a single path; if there are variations, pick one, or add them to a notes section

- Continuously refactor, minimizing documentation that is likely to change
- Passive documentation should focus on *when* and *why* rather than process steps, as these are less likely to change

Automation can free you to do more valuable work and help you to move more quickly. By starting with lightweight documentation and moving incrementally toward automating more steps, you can have automated processes that are repeatable and reliable.

Tags:

agile (/keywords/agile), documentation (/other-keywords/documentation), automation (/other-keywords/automation)

Up Next

Plan for the Year the Agile Way (/techwell-insights/2020/01/plan-year-agile-way)
January 10, 2020

GET TECHWELL INSIGHTS DELIVERED WEEKLY

ALL TECHWELL INSIGHTS BY THIS AUTHOR (HTTPS://WWW.TECHWELL.COM/USERS/STEVE-BERCZUK)

RELATED INSIGHTS (HTTPS://WWW.TECHWELL.COM/TECHWELL-INSIGHTS/2020/01/DOCUMENTATION-AUTOMATION/RELATED-INSIGHTS)

Login (/user/login?destination=node/199469%23comment-form) or Join (/user/register?destination=node/199469%23comment-form) to add your comment

About the Author

(/users/steve-berczuk)

Steve Berczuk (/users/steve-berczuk)

Steve Berczuk is a Principal Software Engineer with experience as a manager, Scrum Master and technical lead in Boston, MA. The author of *Software Configuration Management Patterns: Effective Teamwork, Practical Integration (http://www.stickyminds.com/s.asp?F=S541_BOOK_4)*, he is a recognized expert in software configuration management and agile software development. Steve is passionate about helping teams work effectively to produce quality software. He has an M.S. in operations research from Stanford University and an S.B. in Electrical Engineering from MIT, and is a Certified ScrumMaster. Contact Steve at steve@berczuk.com (mailto:steve@berczuk.com) or visit berczuk.com (http://www.berczuk.com/) and follow his blog at blog.berczuk.com. (http://blog.berczuk.com)

TechWell Insights To Go

Get the latest stories delivered to your inbox every month.

| Your email address | Please select your country ⌄ | SUBSCRIBE |

Conferences (/software-conferences)
Communities (/software-communities)
Subscribe to TechWell Insights (/subscribe-techwell-insights)
Associated Training and Consulting (https://www.coveros.com/)
Advertise (/advertise-with-techwell)
About Us (/about-us)

Join Mailing List (/join-mailing-list)
FAQ (/faq)
Speak at a Conference (/software-conferences/be-a-speaker)
Press & Media (/press-and-media)
Contact Us (/contact-us)
RSS (/rss)

(//twitter.com/TechWell)   ((//www.linkedin.com/company/techwel   (www.hub.techwell.c
https://www.youtube.com/user/SQEVideo)