

5 Ways to Overcome Your Hatred of Test Documentation

By Michael Stahl - March 20, 2017

As published on StickyMinds.com (<https://www.stickyminds.com/print/article/5-ways-overcome-your-hatred-test-documentation>)

Summary:

As far as I know, there isn't yet a support group for testers. But if there were, I believe many of us would confess to having a severe aversion to writing test documents.

Part of my job definition is to be the annoying person who pushes people to write documents. I defined the document templates, and I review most of the test documents my team produces. I can tell you that it's just about as you imagine: unexciting and much less fun than running tests and finding an interesting bug.

Worse, because I am the one preaching about test documents, I have to be a good example and do what I ask others to do. I can't skip writing test documents for any of the projects I am responsible for, even though I *really* don't enjoy it. In fact, when writing test documents, I have to get up every now and then and take a walk in the hallway just to wake up.

Even though it's not the most interesting task, writing test documents is a good practice to have. It enforces an orderly thought process about the test challenge ahead, explains what we're planning and why we do things this way and not another way, and improves the action plan, the test strategy, effort estimation, and task partition.

Documented plans also let stakeholders know what they can expect to get from the validation team. It allows feedback if we made mistakes in our plan and reduces the number of surprises caused by misunderstanding or mismatched expectations.

None of this should be news to you, but it still doesn't make the process any more fun. However, I may be able to make the idea of test documentation a little easier, or at least a little more palatable. You're going to have to do it anyway, so why not try to overcome your hatred of test documentation and try to get in a better mental state about it? Here are five tips.

1. Write the test document for yourself

Get used to the fact that very few, if any, people will read the test document. This is OK. The main beneficiary of a written test document is the author. Writing it helps you order your thoughts and organize the test plan into clear steps you (and anyone else) can understand.

Besides, if you don't expect anyone to read the document, you won't be disappointed when that's just what happens. In their book *Exploring Requirements: Quality Before Design*, Donald C. Gauss and Gerald Weinberg relate to this disappointment. They write, "The document is nothing; the documenting is everything."

Writing the document is the goal. If it is read by others, that's a bonus.

2. Use your human thought process

In today's Twitter world, we feel that any idea can be conveyed in one or two sentences—and if we need more, someone must have already found a trick or developed an app that will do the job for us. We believe that if we only think hard enough, we will find a technological solution that will write documentation for us, like **Doxygen** did for code documentation.

It's frustrating to spend a lot of time doing something manually when all the time you feel it could be automated and be done more efficiently. But writing a document is a *thought activity*, and luckily no one yet has found out how to automate that. So stop torturing yourself about wasting time and revel in the fact that this is a human job that isn't replaceable.

3. Teach yourself touch-typing

Part of the reluctance to write test documents comes from the fact that it takes a lot of time. It takes even more time if you only use two fingers or have to keep stopping to look at the keys.

Touch-typing won't only improve your test documents, but also the clarity of your bug reports, your code documentation, and your emails. When typing is hard, you tend to skip writing or express yourself in one-syllable words. When typing is easy, your prose is flowing, clear, and easy to read.

4. Master the language

For non-English speakers who need to write test documents in English (or any language that is not mastered at the mother-tongue level), the language itself is a barrier to efficient writing. If you struggle to formulate a sentence because the language is foreign, writing takes more time and can be extremely frustrating.

To help fix the process, first, ask if the document *has* to be in the foreign language. If it does, check if you can write in your native language and have the document translated. Your time may be significantly more valuable to your company than the cost of a translator.

However, if you know that the business environment in your country is such that you will always need to work in a foreign language, invest effort in improving your mastery of the language. Here, too, your company may agree to cover the costs.

One way of learning a new language that worked well for me is reading a lot in that language. In high school, I had to improve my English. At about the same time I discovered science fiction. For three years, I almost exclusively read sci-fi in English. Not only do I know **The Three Laws of Robotics** and grok what *grok* means, but I also got much better at writing English.

Find books in the language you need to learn about topics you love, and you will have fun while learning.

5. Learn how to write

I'm not just talking about how to write test documents. Learn to write anything—creative writing, for example. Once general writing is more fluent, writing work-related documents will be easier as well.

In his **must-read article** about the importance of requirements documents, software engineer Joel Spolsky says, "Writing is a muscle. The more you write, the more you'll be able to write." Splosky testifies from his own experience: to overcome his fear of writing, he took a class in college that required him to write an essay every week.

If you're stuck with writer's block when it comes to your test documentation, try writing something—anything—else first. You'll find the words will flow more easily.

Just Do It

There are many things you do in life because you should, not because you want to: eating right, working out, PTA meetings, flossing ... so just add test documentation to this category. And stop procrastinating. My experience is that when you write test documents on time—at the start of a project, not toward its end—in most cases, you will feel the effort was worth it and that you gained some benefit. I hope these tips can help.