Why is software testing so important? How to share the value with your team

Software testing is important – you already know that. But, how do you share its value with the rest of your company?

Let me ask you a question: do you get *real* value out of the testing you do?

Your answer may depend on the team you’re on. If you’re a project manager or a developer, you might think, “No, testing is actually really expensive and time-consuming. Can’t I just ship features already?” But, if you’re a tester, you’re likely to say, “Of course there’s value in testing! But my team seems to be the only one who knows it.”

The problem with sharing the value of software testing

In a recent lecture, I asked a large audience of testers if their organizations value their work. Over 80% of them answered no. With such an overwhelming majority feeling this way, it made me wonder what made them come to this conclusion. How did they perceive “being valued”? And how was this “value” being defined?

**The problem is that value isn’t necessarily an objective measure.**

Sure, there’s things like ROI and attribution models that can determine the *monetary* value of a certain activity, but when it comes to bigger questions like, “Should we invest in QA?” value is often a matter of personal perception and interpretation.

It’s interesting to see the difference between how testers and non-testers value software testing work. Some non-testers may understand this work as looking for worthless bugs that will not be fixed all while wasting valuable time running useless tests. Meanwhile, testers see the value in gathering and presenting information about the product in order to help stakeholders make good decisions. Bugs are a by-product and tests are a necessity, even when it seems like a specific test has a low probability of unearthing a major catastrophe.

As you can see, many times there is an abysmal value gap from team to team, and in my experience this a direct result of lack of communication between them.

Beyond water cooler talk

Sharing the value of your team’s work isn’t just about chatting during lunch. Great communicators know how to share and distribute information in a way that generates real value for all involved. So how can we testers do this? We need to work in a strategic way: collecting data that relevant stakeholders need, organizing information accordingly, and making sure it is being processed correctly.

In many cases, we see that testing data does not become actually valuable mainly because testers fail to communicate the importance of their tasks and the value of their work.

That is where test management tools come in. By sharing your testing metrics and interactive dashboards across your organization, these tools help you structure your communication and convey value.

Your message is important – make sure your data supports it!

As part of our work as testers, we collect data about the product and project status. Some of this information is very important, while other data may not be as relevant to your stakeholders.

As a tester or test lead, it’s important to constantly ask yourself what information your stakeholders need, and make sure your testing operations are gathering this information and processing it.

And, if your data is scattered across multiple sources, then being able to consolidate your data into one place will paint a complete picture that will be easily understood by your busy audience.

For example, a dashboard like the one below presents a host of information that can help you communicate the value of the work being done. In this particular case, the tester has brought together important information about the type of bugs found, the amount of issues the team is working on, the status of each issue, and even who on the team is working on each issue. This sort of clarity into what the team is working on helps provide stakeholders the information they need to value the team’s work.

Interpretation is no less important than the data itself

In my experience, all good communication fulfills a few important criteria. In my work with stakeholders, I like to follow the SMART principles:

* **Simple.** Try to convey your messages using one-liners instead of paragraphs. Whenever possible use graphs instead of words. Assume your reader has little time and a limited attention span.
* **Measurable.** Numbers and statistics speak louder than opinions and descriptions. This doesn’t mean that your opinions are worthless – sometimes with experience you simply know the answer based on your gut feeling – but supporting this feeling with hard data and evidence will make it more credible.
* **Actionable.** Data analysis is most effective when it suggests a solution instead of only pointing out flaws. In many cases, you, as a tester, are the best source of knowledge. Humility is always important, but if you have a solution, don’t be afraid to speak up.
* **Repeatable.** When you are providing information, you will most certainly be asked what was the previous status of the issue. The most common question you will hear is, “Is this a regression?” This means that your checks and tests should be repeatable and you should strive to have historical information whenever possible.
* **Timely.** Provide information while it is still relevant, and when your stakeholders will be able to make good decisions. If you say the right thing, but at the wrong time, it will have little or no impact at all. For example, if you find an issue that will take weeks to solve when you are only days from the release date, it means that your team will either ignore the issue or they will have to delay the release. Keep your release date in mind so that any issues can be addressed before it’s go time.

Remember: your deliverable is information, not testing

Could it be that we, the testers, are the guilty ones for making our organizations undervalue our work?

If all we are doing is running tests, complaining about the lack of testing environments, or simply putting together tables with tons of data that people cannot comprehend, then we are not really helping the process of delivering great products to the field.

It’s about time we all understand that fundamentally, our main job is to help make decisions that will deliver a better product, on time, and within the scope! This is a harder job than simply running tests and reporting issues, but especially in today’s rapid delivery environment, we all are responsible for helping sail the ship in the right direction.

.