

Webinar Highlights: The Tester's Story - Painting the Picture of Risk

In this webinar, Shaun Bradshaw, VP of Consulting Solutions, shares practical experience on how to build and maintain trust with both developers and project stakeholders.

The Tester's Mindset:

Healthy skepticism

- Not pessimism
- Prove that it works

Curiosity

- Ask questions
- Observe how system will react

Emotional Intelligence

- In communication, being able to understand the emotional state of the people you are communicating with as well as your own emotional state

Learning and Observation

- Need to be open to learning
- Adjust how you are testing based on observation

Try to Break the System

- Ability to dive in and figure out what is going on in the system

The Tester's Perspective

Should have a combination of the following skills:

- Analytical
- Technical
- Business Oriented



Champion of Quality:

- Know difference between QA and testing
- Communicate the value of defect prevention and defect detection
- Measure the quality of development processes
- Tester's role is to show risk

Communicating Risk as a Tester

What and how do testers handle communication of risk with:

Other test team members

- Ensure that you don't report a bug that someone else on the team has already reported
- Ensure that you don't report a bug that is really a data set or environmental issue.

Developers

- Be clear as to what risk is when communicating bugs. Severity may not align with priority.

Stakeholders

- Risk from revenue or end user perspective

When and What to Communicate:

Before Coding begins

- Problems with user stories, requirements, specifications
- Backlog grooming, walkthroughs, reviews, collaboration (Agile)

During Coding

- What will be tested
- How it will be tested
- Data requirements
- Finding bugs

Other thoughts about Communicating Risk:



- Shared goal across team-Deliver a quality product
- Positive attitude
- Work to influence developer priorities
- Be willing to negotiate with developers and stakeholders
- Maintain emotional intelligence when communicating risk (verbal and written)

Building trust with developers

- Don't be a Chicken Little
 - The sky isn't always falling. We can't always be in crisis mode.

Don't cry wolf

- Don't document non-issues
- Don't call their baby ugly
 - Emotional attachment to their code
- Talk to them
- Demonstrate a willingness to learn
- Spend time with Developer to find out how functionality will work

Take Responsibility

- If you opened a bug but it was really problem with your test, or data, be upfront about it
- Investigate issues
- Try different value, give a lot of information

What to do when a test fails

If a test fails, is it a defect? Ask the following questions:

- Can failure be duplicated?
- Was the test properly executed?
- Was the failure due to environmental or data issues/configuration?
- What error message was generated?
- What is the nature of the failure and what the potential causes?
 - Do some additional exploratory testing around that problem
- Assume that failure isn't a bug until you can prove otherwise
- Record failure in run log
- Communicate (as directly as possible) with the development team



Document the failure in a defect report

- Include as much detail as possible
- Include results of investigative tests
- Verify that fixes are available for re-test prior to release to production

How to demonstrate risk to project stakeholders:

- Risk is an exposure to harm or loss
- Speak their language (Revenue, Customer, Reputation)
- Understand that the final decision is theirs, and not yours
- How many of these issues were known before the release?
- How many could have been detected with just a few more days of testing?
- Don't let your project stakeholder make the same mistakes (Protect them)

