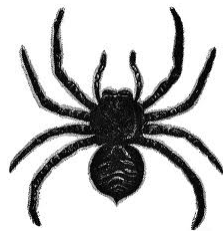# How Did I Miss That Bug?

Overcome Cognitive Bias In Testing

Gerie Owen
Northeast Utilities

Peter Varhol
Telerik

Why are we talking about missing bugs?

- Have you ever missed a bug?
- Have you ever been asked how you missed a bug?
- Have you ever wondered how you missed a bug?

Consequences of Missed Bugs

- Possible Consequences of Missed Bugs:
    - o Negative Publicity
    - o Lost Sales
    - o Lost Customers
    - o Even Loss of Life!

MISSED BUGS CAUSE MAYHEM!

How Do We Miss Bugs?

- Missed test cases
- Misunderstanding of requirements
- Misjudgment in risk-based testing
- Inattention
- Fatigue
- Burnout
- Multi-tasking

What Is a Missed Bug?

An error in judgment

Therefore, to determine how testers miss bugs, we need to understand how humans make judgments, especially in complex situations.

○   ○

How Do We Make Judgments?

- Thinking, Fast and Slow – Daniel Kahneman

  o System 1 thinking – fast, intuitive, and sometimes wrong

  o System 2 thinking – slower, more deliberate, more accurate

○   ○

How Do We Apply System 1 Thinking?

- Keeps us functioning in a complex world
  - o Our initial reactions to situations.
  - o Fast decisions, usually right

  - o May employ heuristics or rules of thumb
  - o Can be gullible and biased

How Do We Apply System 2 Thinking?

- System 2 makes deliberate, thoughtful decisions
  - o It is in charge of doubt and unbelieving
  - o Is applied when we analyze a problem
  - o Often lazy
  - o Difficult to engage

The Origin of Biases

- System 1 and System 2 can be in conflict:

  o This leads to **biases** in decision-making

How do Biases Affect Judgment?

- We maintain certain beliefs
  - o Which may or may not be factually true
  - o Those biases may influence our decisions
  - o We may be predisposed to believe something that affects our work and our conclusions

How Do Biases Affect Testing?

o We may test the wrong things

o We may miss test cases

o Not find errors, or find false error

o We may test by rote

o We may become fatigued and succumb to cognitive strain.

The Representative Bias

We judge the likelihood of an occurrence in a particular situation by how closely the situation resembles similar situations.

. Testers may be influenced by this bias when designing data matrices.

The Curse of Knowledge

We are so knowledgeable about something that our ability to address it from a less informed, more neutral perspective is diminished.

When testers develop so much domain knowledge that they fail to test from the perspective of a new user.

The Congruence Bias

Not considering alternative hypotheses

Testers write test cases to validate that the functionality works according to the specifications

They may neglect to validate that the functionality doesn't work in ways that it should not.

The Confirmation Bias

The tendency to search for and interpret information in a way that confirms one's initial perceptions.

Testers' initial perceptions of the quality of code, the quality of the requirements and the capabilities of developers can impact the ways in which they test.

Missed negative test cases can be the result of this bias as testers seek to prove what they believe to be true without trying to disprove it.

The Anchoring Effect

rely too heavily on one piece of information exclude other ideas or evidence that contradicts the initial information

Software testers do this often when they validate code to specifications exclusively without considering ambiguities or errors in the requirements.

The Anchoring Effect

Software testers do this often when they validate code to specifications exclusively without considering ambiguities or errors in the requirements.

○ ○

Inattentional Blindness

A psychological lack of attention

The tendency to miss obvious inconsistencies when focusing specifically on a particular task.

○ ○

Inattentional Blindness

This happens in software testing when testers miss the **blatantly obvious** bugs

Why do we develop biases?

- The Blind Spot Bias

    o We evaluate our own decision-making process differently than ~~~~ uate how others make decisions.

    West, Meserve and Stanovich

How Does This Apply To Testing?

- We must manage the way we think throughout the test process.

  o As Individual Testers
  o As Test Managers
  o As a Professional Community

How Can Testers Manage Their Thought Processes?

- Use more System 1 Thinking?
  - OR
- Use more System 2 Thinking?

Test Methodology and System 2 Thinking

Test methodology is the analytical framework of testing

It naturally invokes our system 2 thinking and places the tester under cognitive load.

○   ○

Test Methodology and System 2 Thinking

The determination of whether the actual results match the expected results becomes an **objective** assessment.

But where is the **subjective** assessment?

○   ○

How do we find the bugs?

Focus on System 1 Thinking,
Intuition and ***Emotion***

Focus On System 1 Thinking

- Heuristics used with Oracles

- Recognize our emotions as indicators of potential bugs

- Exploratory Testing

How Should We Use Exploratory Testing?

- *Unstructured*
  - Before beginning test case execution
    - Minimizes preconceived notions about the application under test
  - Oracle Based
    - Users' perspectives
    - Data flow

How Should We Use Exploratory Testing?

- *Structured*
  - Use to create additional test cases
    - May be done earlier, possible as modules are developed
  - Session-Based
    - Time-Boxed Charters
    - Multiple Testers
    - Post test review session

What Can Test Managers Do?

- Foster an environment in which the testers feel comfortable and empowered to use System 1 thinking.

  o Plan for exploratory testing in the test schedule
  o Encourage Testers to take risks
  o Reward for **Quality** of bugs rather than **Quantity** of test cases executed

What Can the QA Profession Do?
A Paradigm Shift

  o Shift our focus from requirements coverage based test execution to a more intuitive approach
  o Exploratory testing and business process flow testing becomes the norm rather than the exception
  o Develop new testing frameworks where risk-based testing is executed through targeted exploratory testing and is balanced with scripted testing

How Do We Find The Obvious Bugs?

FOCUS LESS

Use Intuition

Believe What We Can't Believe