The Evil Tester's Guide to Technical Web Testing

Let's Test 2013

Alan Richardson
@eviltester

www.eviltester.com
www.compendiumdev.co.uk
www.seleniumsimplified.com

I assert that Tester’s need technical skills when testing ‘anything’. Since I primarily test web applications, I will describe my technical view of Web Testing.

In this session I will describe some of the tools that I use when testing, not because technical web testing revolves around tools, but because the technical testing thinking process leads to the finding and utilisation of tools.

Tools are the draw, the quick win for attendees. But thought processes sit at the core of this talk. They lead to tool augmentation, to exploring the built in capabilities of the browsers, and to the skills we need to test technically.

I will talk a little around theory, and, dangerously for a live session, I will provide a demonstration of tool augmented technical testing.

@EvilTester

Technical Testing
What is Technical Testing? To You?

• How does the phrase make you feel?

• To you as a manager?

• To you as a tester?

• Can you identify with it?

@EvilTester

What is Technical Testing? To Me?

- A reminder to keep "going deeper"
- It means "Tool Augmentation"
- How I describe what I do

@EvilTester

A reminder to keep going deeper

MORIM - Model through Observation, Reflection, Interrogation and Manipulation

- Model == what I think I currently understand
- Observations can corroborate or invalidate my model
- Reflect to find gaps and lack of depth and derive intent
- Interrogate - focused observation with intent
- Manipulation - hypothesis exploration

@EvilTester

It means "Tool Augmentation"

- Tools to passively observe, maintain history of observations
- Tools to alert on specific conditions
- Tools to observe the unobserved, and interrogate the inaccessible
- Tools to help me model and reflect
- Tools to help me manipulate
- ... etc.

Never tools to control. Tools to augment.

@EvilTester

How I describe what I do

- Not a definition
- A description of my current approaches
- I try get as deep and technical as I can
- I need to keep learning so that I can understand the technology
Go beyond the surface structure

- Transformational Grammar
  - Surface & Deep Structure

- Questions operate as tools to investigate Surface to Deep mapping in people

- Bug?
  - Surface \(!=\) Deep
  - Surface_1 \(!=\) Surface_2

@EvilTester

Transformational Grammar

- Terminology originated with Chomsky
- Multiple surface structures
- Single Deep structure
- Deep structure
  - filtered, biased, distorted
  - Surface Structure

NB: I'm reinterpreting this. Abusing it for my own purpose- to model Technical Testing

@EvilTester

Questions operate as tools to investigate Surface to Deep mapping in people

"Login doesn't work"

- How Specifically?
- How do you know?
- Ever?
- ... etc.

We ask the system questions.
Some ways we notice bugs?

- Surface $\neq$ Deep
  When we can't transform from our Deep Structure to the observed Surface Structure

- Surface$_1$ $\neq$ Surface$_2$
  When we spot different surface structures than we would expect from our transformations

  Surface $\equiv$ Model, Deep $\equiv$ Model, Model $\neq$ Reality

@EvilTester
How to do Technical testing?

- Identify tools to work with System Surface Structures
- Questioning Systems at different surface levels
- Learning System Structure Technology
- Modelling System Surface Structures
Technical Web Testing
Technical Web Testing

● Understand and model the Project context
  ○ scope of testing, aims of testing, stakeholders, etc.

● Understand and model the System context
  ○ the web context
  ○ the application context
  ○ the deployment context
  ○ the usage context

● For each identify
  ○ Technical Risks
  ○ Test Ideas

Technicalities for Web Testing

● Understand and model the System context
  ○ the web context
  ○ the application context
  ○ the deployment context
  ○ the usage context

● For each identify
  ○ How can we observe it?
  ○ How can we interrogate it?
  ○ How can we manipulate it?
    • Summary, Log, etc.
    • Details & Different views
    • Change & Amend
Web Context: A Browser View

Browser

HTML Page

JavaScript

CSS

HTML

Objects: Flash, ActiveX, Silverlight, etc.
Web Context: A Browser View - Technical Risks
Web Context: A Browser View - Technical Tooling Augmented
Exercise: Let's Elaborate this Model
Exercise:
What other models might we build?
Model suggests... Tool suggests...

We can work from model to tool...

We can also work from tool function to model.
"All together now" - a tool augmented group exploration demo

- Using only out of the box browser functionality.
- When exploring a web app
- What do we get out of the box to observe, manipulate, interrogate?
- What can't we do?
- What tools might let us do it?
- What does the tool functionality suggest for us?

@EvilTester

Summary

- Technical Testing works with multiple models of the System
- Model for:
  - Understanding
  - Risk
  - Test Ideas
  - Tooling ideas
- Tool Augmentation
  - Observe
  - Interrogate
  - Manipulate
Alan Richardson is an Independent Test Consultant based in the UK. He offers training and consultancy in Selenium WebDriver, exploratory and technical web testing.

- [uk.linkedin.com/in/eviltester](http://uk.linkedin.com/in/eviltester)

Contact Alan for training and consultancy tailored to your needs: [alan@compendiumdev.co.uk](mailto:alan@compendiumdev.co.uk)

**Blogs and Websites**

- [SeleniumSimplified.com](http://SeleniumSimplified.com)
- [EvilTester.com](http://EvilTester.com)
- Testing Papers and Tools
  - [CompendiumDev.co.uk](http://CompendiumDev.co.uk)

**Online Training Courses**

- Technical Web Testing 101
  - [Unow.be/at/udemy101](http://Unow.be/at/udemy101)
- Intro to Selenium
  - [Unow.be/at/udemystart](http://Unow.be/at/udemystart)
- Selenium 2 WebDriver API
  - [Unow.be/at/udemyapi](http://Unow.be/at/udemyapi)

**Videos**

[youtube.com/user/EviltesterVideos](http://youtube.com/user/EviltesterVideos)

**Books**

Selenium Simplified

[Unow.be/rc/selsimp](http://Unow.be/rc/selsimp)