An Introduction to Customer Focused Test Design

Alan Page
Microsoft
@alanpage

http://angryweasel.com/blog
What is Test Design?
Does it work?
Does it have value?
Are you satisfied?
Are your customers satisfied?
Functionality First?

Why are we looking for functional bugs? They're easy. Stop looking for functional bugs. @docjamesw #StarWest

“…testing, for a long time and in many places, has been myopically focused on functional correctness, rather than on value to people.”

-Michael Bolton
Random Internet Quotes

“I don't care about features or plug-ins or add-ons or whatever. Just load the page quickly and let me get on with my work”

“…the voices calling "I don't care about features, just fix the bugs" were louder than ever”

“<some company> makes wonderfully full featured and functionally correct software, but my wife can’t figure out how to use it”
But it’s not like we *ignore* the customer...
Approximately 10% of customers have HD or higher resolution.

68% of customers with a 1600x1200 display lower their resolution.
AshLaws: @soapsuds You should check out Microsoft Office OneNote, it's screen clipping and notes organising is second-to-none.

AshLaws: less than a minute ago.

mbachrodt: Lots of good training of teachers, today; OneNote, PowerPoint, & Excel. It was great to see wide, absorbing-it-all eyes. Wheels are turning!

mbachrodt: less than a minute ago.

msrberlin: New blog post: Canvas for OneNote

http://tinyurl.com/cfqtx8

msrberlin: less than a minute ago.

Synaloep: Definitely loving OneNote. And 1p flights (unsurprisingly, a cheeky tax on it. Cheers Ryanair). Applying to TU Braunschweig soon, scared.
Reactive

Proactive!
Does the software help the customer do anything useful?
Customer Focused Test Design
Not You

You

Copies may not be made or distributed for commercial use
Excerpt from PNSQC 2011 PNSQC.ORG
Customer Focused Test Design

Scenario Testing

Shifts & Sparks

Outside-In Testing

Live Testing
Scenario Testing
Scenario Testing: Concepts

Scenarios
Personas
“Real World” Projects
Tina is a fast-rising sales manager at a Fortune 500 company. She manages a team of eight people who work at several sites across North America, Europe, and India. In order to manage her team and help them meet and exceed their sales quotas, she wants to be able to connect with her team frequently. On most occasions, a short message to her team is adequate, but when context dictates, she often needs to speak or see employees who are not co-located with her - not being able to efficiently connect with her team when needed is frustrating. She knows that teamwork is critical in order for her whole team to be successful, and she wants to build a culture of sharing and teamwork in order to enable success.
Personas - Meet Ellen
“Thinking up new & creative ways of finding bugs keeps my job interesting”

Testers are focused on finding bugs, not running and reporting tests. Testers use their skills to search the space of potential bugs creatively and effectively. The thrill of the ‘hunt’ provides them with the motivation to keep going.

Ellen the vigilant tester
“Developers may take the happy path through a feature but I want to think about every other path that can be taken”

Ellen the vigilant tester

Testers and developers approach their work very differently. Testers test the scenario while developers test the code. Testers love finding bugs whereas developers hate finding bugs. Testers think about how the users work, and developers think about how the code works.
Ellen the vigilant tester

Background & Motivations
- Typically does not have any formal CS training or background
- Quite often was once a user of the product being tested
- Sense of achievement comes from making users' lives easier
- Feels like the customer advocate
- Likes tracking down hard-to-find bugs
- Competes with developers
- Has no desire to become a developer

Focus & Approach
- Driven by requirements
- Assessing quality
- Scope is user centered and scenario driven
- Often exploratory or ad-hoc
- Likes to be able to spend time with users
- Finding bugs, not running tests
- Tests defensively
- Provides detailed and accurate bug reports
- Dislikes ambiguity

Authoring Tests
- Automation for repetitive tasks
- Elucidate designs
- Hands-on approach to catch unforeseen bugs
- Doesn’t believe in 100% automation
- Wants tools that help find bugs
- Creates flexible, reusable tests
- May not always use formal test cases to find bugs
- Always looks for opportunities to find bugs
“Real World” Projects

Goals:
- Build customer inspired solutions that span critical scenarios
- Identify scenario gaps and functionality bugs at the intersection between applications
- Build strong cross-team relationships

Execution:
- Build “real” customer solutions (make heavy use of scenarios)
Outside-In Testing
The standpoint of a given module, the development process, can be characterized as shown in figure 5.

**FIGURE 5. Outside-in Design: Inside-out Testing**

generally proceeds from the global to the local level. The design begins at the system level and becomes more detailed until it is sufficiently detailed to be coded. Once coded, the module is verified in an operational environment.
Performance, Security, Reliability, etc.

Unit Testing

Functional Testing

Component Testing

Feature Testing

Integration Testing

System Testing

“Other”

Performance, Security, Reliability, etc.
System Testing

Integration Testing

Feature Testing

Component Testing

Function Testing

Unit Testing

“Other”

Performance, Security, Reliability, etc.
The Cost of Bugs

Cost ↑

Time →
Is The Cost Curve Always True?
Hypothetical Cost Curve – Design Bugs

Cost $\uparrow$

Time $\rightarrow$

Copies may not be made or distributed for commercial use

Excerpt from PNSQC 2011

PNSQC.ORG
Hypothetical Cost Curve – Functional Bugs

Cost $\uparrow$

Time $\rightarrow$

Cost $\uparrow$
Example Reliability Test

Send Web Query 10,000 times

Or

Send malformed web query 10,000 times
Example Performance Test

Test timing / latency of an action or scenario

Or

Test large number of simultaneous connections
Example Usability Test(s)

Look & Feel

Text & Control alignment

Accessibility tests
Live Testing
But *our* application isn’t a service...
Testing (not) in Production ideas…

Dogfood

Data, data, data

User testing

Frequent updates

A/B Testing
Shifts & Sparks
SHIFTS: Macro Forces of Change

Lens: Information work and business productivity

“STEEP”: Social, technological, economic, environmental, political landscape

Timeframe: over the next 5-10 years (2-3 product cycles out)
Think Ahead…

Ask:

“What tests can I think of in areas that may be affected by future ‘shifts’”? 
Shifts & Sparks - Example

Area: VOIP Call

Shift: Emerging markets – inconsistent network access

Test “Idea”: Simulate varying (and dropped) network access during voip call
Shifts & Sparks - Example

Area: Video Enhancements

Shift: Mobile Computing – multiple devices

Test “Idea”: Simulate testing with, and moving between multiple video devices for video playback
Parting Thoughts & Questions