Building on Success
Beyond the Obvious

Erik van Veenendaal
www.erikvanveenendaal.nl
PNSQC 2020
No More Testers!
No More Testers!
No More Testers!
No More Testers!
Common Testing Challenges

- Increasing business importance
- Digital transformation
- Increasing code size and complexity
- Technology advancement
- Systems-of-systems
- Variety of devices and OS’s
- Security vulnerabilities
- Outsourcing
- Time-to-market critical

Number of defects hardly decreases
Problemen kernreactor Doel veroorzaakt door softwarefout

Terugkeren Boeing 737 MAX mogelijkt vertraagd na nieuwe softwarefout

Duizenden gevangenen te vroeg vrij door softwarefout

Geen melding van schadeoffers vanvrijsluiting daders door softwarefout

Handelaren Amerikaan een huis gezaagd na softwarefout van bank

Tiedelijk onoplossbare zelfstijgende lift te wijten aan softwarefout

Oudere iPhones zijn bijna onder de softwarefout groothandel

De problemen bij een kernreactor in Doel, vlakbij de Nederlandse grens boven Achtstanden, werden veroorzaakt door een softwarefout.

De storing van woensdag in het netwerk van KPN is waarschijnlijk veroorzaakt door een softwarefout. Het vermoeden daarvan is "zeer sterk", zegt een woordvoerder.

Een gevangenis in de Amerikaanse staat Washington heeft ruim 2200 gevangenen te vroeg vrijgelaten door een softwarefout.

Een softwarefout bij de Amerikaanse bank Wells Fargo heeft er mogelijk toe geleid dat vierhonderd Amerikanen uit hun huis zijn gezet.

De bank biedt een overkantingprogramma aan, waarbij hypotheekkredietlijnen worden geheel vrijgegeven.

Een softwarefout bij spoolheten en ziektenhulpen door softwarefout groothandel

Oudere iPhones zijn bijna onder de softwarefout groothandel
Agile ≠ Silver Bullit

Top 5 Agile Benefits

1. Ability to manage changing priorities
2. Improved project visibility
3. Delivery speed/Time-to-market
4. Increased team motivation
5. Increased team productivity

Source: 14th Annual State of Agile™ Report

“despite continuous growth in adoption of agile, many organizations are still facing the same challenges as in previous years”, World Quality Report
Common Testing Practices

Many organizations:
- No test strategy
- Useless test plans
- No test techniques applied
- Few have effective reviews
- Test automation not successful
- Re-usable testware hard to find
- “Testers” not trained for the job
Paradox

- Systems are still released!

- How to be “good enough” and “survive”

A tool box for agility / lean in testing!
Manifesto and Values

- Individuals and interactions
- Working software
- Customer collaboration
- Responding to change

over
- Process and tools
- Comprehensive documentation
- Contract negotiation
- Following a plan

Communication
Simplicity
Feedback
Core Practice #0
Core Practice #1: Risk-Based

"Testing is risk-based"

Define Clear Testing Priorities

- Communication, Customer Collaboration
- Working software & Responding to change

- Product Risk Session
- Risk-Based Test Approach
- Risk-Based Tracking (business language)
Setting Priorities

- Risk identification
  - Requirements (user story) based
  - Brainstorming, failure history

- Risk analysis
  - Risk = Impact x Likelihood
  - Much better ...... Risk = f (Impact, Likelihood)
Product Risk Matrix

- **Likelihood**
  - Low: 3
  - Medium: 5
  - High: 1

- **Impact**
  - Low: 2
  - Medium: 4

- **Business Risk**
  - Technical Risk

- **Risk**
  - Low: 3
  - Medium: 5
  - High: 1
Test Approach

### Test design techniques
- Review software design
- Support unit tests
- Review unit tests

### Exploratory testing
- Support unit tests
- Review unit tests

### Medium Risk
- 5
  - Test design techniques
  - Review software design
  - Support unit tests
  - Review unit tests

### Low Likelihood
- 3
  - Exploratory testing

### Medium Impact
- 4
  - Test design techniques
  - Review software design

### High Risk
- 1
  - Exploratory testing
- 2
  - Test design techniques
  - Review software design
Test Plan

[Diagram showing a risk matrix and test approach sections labeled Mo, S, Co, and W]
Core Practice #2: Reviews

“Testing is preventing defects”

Perform Limited Effective Reviews

- Priority to Risk
- Review practices that make the difference
- “Agile inspection”

Communication & Feedback
Customer Collaboration
It’s Simple but Not Easy

- Different review types for different objectives
- Trained moderators
- Apply roles, checklists and kick-off
- Entry check
- Not too many pages (± 20)
- Good logging rate (2 per min.)
- Reasonable checking rate (± 10 pages)
Core Practice #3: Unit Testing

“Testing requires independence”

Get Developers doing Unit Testing

- Code coverage
- Buddy Testing (XP)
- Test Driven Development
Unit Testing Process

most high level testing will only achieve 30 - 40% code coverage
Continuous Integration & Testing
Core Practice #4: Test Design

“Testing is finding defects”

Use Test Techniques as Tools

- Applying Test Design (start-up)
- Classification Tree Method / Use Cases
- Exploratory Testing

Responding to change, Customer collaboration and No comprehensive doc’s
Apply Test Design

- Identification test conditions
- Establish traceability to requirements

<table>
<thead>
<tr>
<th>Terms</th>
<th>Valid values</th>
<th>Invalid values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Input voltage (Vin)</td>
<td>Default, real (unit / engineering notation), &gt;=0</td>
<td>Empty, not numeric, &lt;0</td>
</tr>
<tr>
<td>Input resistance (Rin)</td>
<td>Default, real (unit / engineering notation), &gt;=0</td>
<td>Empty, not numeric, &lt;0</td>
</tr>
<tr>
<td>Coil inductance (Lcoil)</td>
<td>Default, real, &gt;0</td>
<td>Empty, real, not numeric, &lt;=0</td>
</tr>
</tbody>
</table>
Test Design Start-up
Exploratory Testing Process

Session-Based Testing

Charter

Keep track of bug vs. fix ratio!

Heuristics
- Boundaries
- CRUD
- Configurations
- Interruptions
- ....

Sessions
- Exploration
- Design
- Execution

Debriefing
Core Practice #5: People

“Testing is an extremely complex, intellectual challenging task“

Build Experienced and Skilled Testers

- Training (on-the-job)
- Team Dynamics
- Not only for Testers!!

Individuals and interactions over processes and tools
Testers’ Skills - Not only for Testers!

- Domain
- Development
- Soft Skills
- Testing

Ability to work outside of core area

Functional area, discipline, or specialty

via Sharing Experiences, Best Practices, Coaching Workshops and Training
More People Factors

- Job satisfaction
  - Getting respect & feeling valued, adequate rewards, etc.

- Test guild
  - Test meetings, best practices, etc.

- Career paths
  - Job description & training, promotion, etc.

- Team Dynamics
  - Effective teams don’t just happen
  - Correct skills, Correct attitude, Common goals
Lean Testing Tool Box

- Clear Testing Priorities
- Effective Reviews
- Developers doing Unit Testing
- Test Techniques as Tools
- Experienced & Skilled Testers
No More Testers, or ...

There has never been a time when software testers were needed more!
THANK YOU!
www.erikvanveenendaal.nl

erik@erikvanveenendaal.nl

@ErikvVeenendaal