Narrative Testing: Tools for Story

Test-Driven Development

Mickey Phoenix
Senior Software Development Engineer, Certified ScrumMaster
SolutionsIQ

Rand Huso
Senior Software Development Engineer, Certified ScrumMaster
SolutionsIQ
Mickey Phoenix

- Specializes in narrative testing, test-driven development, automated testing, Scrum, Agile software development practices, and troubleshooting/debugging
- Recently spoke at SD West 2008 on Domain Specific Testing Languages and Agile Usability
- Senior Software Development Engineer at SolutionsIQ
- Certified ScrumMaster
- Email: MPhoenix@SolutionsIQ.com
Rand Huso

• Over 20 years of software development experience in such roles as Software Architect, Team Lead, Senior Developer, and Lead Designer
• Began using Object Oriented methods in 1988
• Senior Software Development Engineer at SolutionsIQ
• Certified ScrumMaster
• Email: RHuso@SolutionsIQ.com
Narrative Testing: Tools for Story Test-Driven Development

- Narrative Testing leverages script-based testing tools and DSTLs to express Story Tests in the user’s own language.
- Developers and customers cooperate to create customer-readable scripts based on User Stories, which are executed interactively against the application UI.
- Through Narrative Testing, customers are able to:
  - view the executing test
  - tie steps in the test to changes in the application state, increasing customer confidence in testing.
- This session will share practical tips and real-world experiences for the interactive creation of executable Narrative Tests with the user.
Agenda

- Acceptance Testing State-of-the-Art ("story testing"/STDD)
- Defining Narrative Testing
- Demo of STDD
- Why Use Narrative Testing?
- Demo of User Scripts and Narrative Test-Driven Design
The Role of Acceptance Testing

- “Definition of Done”
- “Executable Documentation”

Acceptance Testing helps us build what the user asks for.
Standard Acceptance Tests

- Are written by developers in developer language (often code)
- May not interact with the UI at all
- Are excellent for validating tabular business logic
- Are often presented as input-output tables (e.g. FIT tests)
Story Test-Driven Design

• Organizes acceptance tests by the user story each validates
• Guarantees coverage by tying tests to stories
• Doesn't change the way tests are written
  – Business analysts must translate user requests
  – Test details are hidden from the customer
Building What the User Needs

• Not just asking "What do you want?"
• Not just building what we think the user wants
• Finding a process that draws out the user's actual needs

How do we elicit accurate customer requirements?
What is Narrative Testing?

- Uses stories (User Scripts) to understand customer needs
- Expresses tests in the Customer’s Language
- Shows the customer visible execution in the application under test (AUT)

Narrative Testing builds
what customers need
Narrative Testing Tools

CubicTest, Selenium, StoryTestIQ, WatiN, Watir

• Some provide Domain Specific Testing Language support
• Some provide test component re-use/hierarchical decomposition
• All provide visible test execution
Narrative Testing Provides

- Accurate extraction of customer needs via stories
- Expression of customer requirements in customer terms
- Visible confirmation to the customer of the execution of their intent
How Do We Build Trust

- Customer confidence through visible execution
- Information, in customer terms, beyond “the green bar”
STDD Demo Using FitNess/FIT
Narrative Testing Demo Using Selenium
Narrative testing is based on the human knack for telling stories.
Are User Stories Enough?

- Better than use cases
  - Users think in narratives, not in decision trees
- Insufficient detail
- A “promissory note for a conversation,” not the conversation itself
- Not executable
User Scripts Are

• Detailed versions of User Stories
  – “the story behind the User Story”
• Written by the customers, in their own language
• Linear interactions between the User and the Software
• Sufficiently detailed to be executed
Programmers and Customers Think Differently

- Programmers are trained in system analysis and think in branching conditionals and truth tables
- Customers are trained in their domain and think in stories
Story Testing

- Customer writes a User Story ("case")
- Developer talks to user to get details ("conversation")
- Developer translates details into a FIT-style test
- Customer approves the test ("confirmation")

This process involves three translations.
The Three Translations

• Customer language to developer language
• Customer narrative to business logic details
• Tabular test whose implementation is hidden to customer understanding
Translation is Lossy

- Every translation step loses information
- Translation between different ways of thinking is especially lossy
- Stories can bridge the gap

“saudade”
Definition...

- "The famous saudade of the Portuguese is a vague and constant desire for something that does not and probably cannot exist, for something other than the present, a turning towards the past or towards the future; not an active discontent or poignant sadness but an indolent dreaming wistfulness."
  - "In Portugal" by A. F. G. Bell
...vs. Story

• “No matter where I wander I'm still haunted by your name
  The portrait of your beauty stays the same
  Standing by the ocean wondering where you've gone,
  if you'll return again
  Where is the ring I gave to Nancy Spain?”
    • “Nancy Spain” by Barney Rush

• “Next year, in Jerusalem.”
  • Traditional ending of the Jewish seder
How Do We Eliminate These Translations?

- Write tests in the customer's language
- Write tests in narrative form, not as business logic definitions
- Show the customer the execution of the test, not just the static script
In the Customer’s Language

• Support domain-specific languages
• Support multi-leveled abstractions (sub-tests, custom commands, etc.)
• Read as much as possible like natural language
  – Linear
  – End-to-end
  – Describe interactions, not outcomes
Visible Execution

• Run against the UI as seen by the customer
• Display each test step’s effect on the application
• Execute the test at the customer's pace (pause/resume, etc.)
Narrative Testing is Like Writing a Play
The Customer as Playwright

- Writes the plot
- Writes declaratively

- Has a vision, but not the details
Tests as the Script

- Describe actions and outcomes
- Provide the structure around which the director elaborates
- Must be performed as written
The Developer as Director

- Adds "look and feel," polish, and details
- Translates declarative intent into instructions to the actors
- Identifies practical limits
The Script is Not the Play

Reading the script and watching a good production of the play are completely different
Seeing “What Works”

- Seeing a dress rehearsal clarifies the differences between imagination and implementation
- Even a bad initial production generates new ideas
- “I know it when I see it” – in a play or in an application
Narrative Testing in Practice

What does Narrative Testing look like in the context of a real project?
Each Iteration Starts with a Working Session

- Customers elaborate user stories into user scripts
- Developers pair with customers to write the user scripts into tests
Cadence

Cadence is a familiar concept from Unit Testing TDD:

– Red/Green/Refactor (cadence)
– This works because the tests:
  ➤ are small
  ➤ can be written incrementally
  ➤ can be written solely by the developer
Cadence in ATDD?

• **Acceptance tests:**
  – are large
  – must be written at the start of the sprint
  – need to be written by (or with) the customer

• **Two cadences instead of one**
  – ‘Run/Code/Run Further’ replaces ‘Red/Green’
  – ‘Finish a User Script/Refactor’ replaces ‘Refactor’
As Each Story is Finished

- The customer is called back to watch the play "in workshop"
- The customer can modify the user script if it doesn't match their intent
At the End of the Iteration

• The "demo" consists simply of running all of the tests for the customer
• The acceptance tests are the customer's user scripts directly, so test coverage is assured
Customer Trust

• Customers can't trust the little green bar (however optimistic it looks) until they are given reason to trust it

• Narrative Testing Tools build trust by showing the customer the play instead of just the script
Narrative Testing Demonstration
Questions?
Further Resources

• Narrative Testing tools:
  • http://storytestiq.solutionsiq.com/
  • http://selenium.openqa.org/
  • http://wtr.rubyforge.org/

• Our website:
  • http://www.narrativetesting.org/
The authors would like to thank SolutionsIQ of Redmond, WA and STSII of Dublin, CA for their generous support of this presentation.
Architecture in an Agile Organization

SolutionsIQ experts share their experiences and practical approaches to better align businesses with architecture goals while adhering to Agile principles.

— Chris Sterling, Principal Consultant, Certified Scrum Trainer and Agile Coach

Domain-Specific Testing Languages (DSTLs)

DSTLs help you keep the core testing tool simple while creating automated test scripts the customer can easily read, verify, and use as requirements.

— Rand Huso, Senior Software Development Engineer

Panel Discussion: Troubleshooting Distributed Agile Team Projects

Leading Agile experts Esther Derby, Hubert Smits, Tamara Sulaiman, Samir Shah join Monica Yap to share their experiences working with distributed Agile teams.

— Monica Yap, Engagement Manager, ScrumMaster, Agile Coach

Punctuated Continuity: Using Ritual and Ceremony to Avoid Process Fatigue

Learn techniques that can be employed to keep repetitive Agile routines invigorating, pulled from actual experiences with teams practicing XP and Scrum.

— Michael Tardiff, Agile Team Lead and Coach
Swarming: The Birds and the Bees and Agile
Discuss the fascinating set of swarming behaviors in the animal world that resonate strongly with some of the central tenets of Agile development.
– Dhaval Panchal, Agile Coach, Analyst, Certified Scrum Practitioner

Assembling a Real-Time Collaborative Development Platform in the Cloud
SolutionsIQ CEO demonstrates a whole platform for Agile development featuring mashups of SaaS and open-source tools for development and continuous integration
– Charlie Rudd, Chairman and CEO
Thank You!

- Come to the SolutionsIQ booth at Agile 2008
  - *Pick up a free Agile t-shirt, and*
  - *Schedule one-on-one sessions with SolutionsIQ speakers*

- Visit [solutionsiq.com/agile2008](http://solutionsiq.com/agile2008) for additional Agile 2008 materials and related content from SolutionsIQ

About SolutionsIQ

*SolutionsIQ offers a full spectrum of services to develop software and fulfill technical talent needs, while improving your Agile knowledge and capabilities. Clients include AT&T (Cingular), Amazon, Corbis, Expedia, Federal Home Loan Bank, InfoSpace, Key Bank, Nike, Nordstrom, Regence Blue Shield, Safeco, US Bank, and Washington State University. A Microsoft Gold Certified Partner, SolutionsIQ is also a member of the Java Community Process, Scrum Alliance, Software Association of Oregon, and Washington Technology Industry Association. Learn more at [www.SolutionsIQ.com](http://www.SolutionsIQ.com).*