ATDD and Continuous Regression Testing with Jubula

Alexandra Schladebeck
Agenda

► Where is the problem?
► What are the possible solutions?
► How can Jubula fit in?
► Demo
Where’s the problem?
Communication

1. How the customer explained it
2. How the Project Leader understood it
3. How the Analyst designed it
4. How the Programmer wrote it
5. How the Business Consultant described it
6. How the project was documented
7. What operations installed
8. How the customer was billed
9. How it was supported
10. What the customer really needed
Signal-loss in development

- Requirements
- Functional specification
- Technical specification
- Implementation
How can we stop this happening?

- Whole team approach
  Integration of developers and testers into one team
- Provide acceptance criteria from outset
  Customer perspective
- Quick feedback
  Opportunity to react

If we know what will be tested, we know how to build it...and we can check that we did it right
Test driven development

Write test

Refactor

Write code
Test driven development

Input: what to test (what is required)

Write test

Refactor

Write code
"Never give away what you test otherwise developers will write the system to pass the tests"
ATDD: Shepherd or driver?
ATDD through the GUI

- Only view customer has of the software
- All layers tested at once
  - Workflows
  - Business processes
- How can we write a GUI test for a GUI that isn’t there?
- How can users understand my tests?
- How can the tests keep up with changes?
How Jubula and GUIDancer fit in

Early test creation
Focus on communication
Best practices

- **Jubula**
  Test creation, execution, analysis

- **GUIDancer**
  Jubula plus: Mylyn, BIRT reporting, Code Coverage, Teststyle, Dashboard
Early test creation

- **Not capture-replay**
  No AUT required

- **Object mapping**
  Test <-> AUT
Communication

- Questions and reactions during test dev
- Tests written from user / tester perspective
  No code
  Focus on readability
  Understandable by business users / customer
Easy maintenance

- Same best practices as software development
  - Reference, don’t copy
  - Central changes to update multiple references

- Central object map

- Data can be centralized

Modular
Flexible
Reusable
"As a user of the DVD Tool, I want to be able to remove categories I have created"

1. Specify test
2. Add data
3. Perform object mapping
4. Execute test
From acceptance to regression

Test automation

INCUBATION (Acceptance)

FULLTEST (Regression)

Testing in agile processes

- Requirements
- Design
- Test specification
- Component test/Implementation
- Integration test

- Daily build
- Regression testing (automation)

n
n+1

ATDD with Jubula  27.03.2012
Demo

What happens if the software changes?
Objects
Data
Workflows

What happens if the software breaks?
ATDD means listening to results

GUILdancer® Report: Test Execution History

Test Execution History

Test Steps (m)
Constant quality information

- From correct perspective

![Diagram showing quality over time with iteration and pre-release test phases.]

- Y-axis: Quality
  - 100%

- X-axis: Time
  - Iteration
  - Pre-Release Test

Legend:
- Orange: automated tests
- Blue: non-automated tests
Thank you!

Further information
http://www.eclipse.org/jubula
http://www.guidancer.com

@EclipseJubula
Further info

Community
- Forum
- Webinar
- FAQs
- Videos
- Documentation
- Mailing lists

Professional services
- Proof of Concept
- Appetizer packages
- Training
- Workshops
- Support
- Test reviews
- Best practices
- GUIDancer Demo Licenses
- Customized extensions
Finally...

- **Eclipse Testing Days 2012**
  - Call for Sponsors
  - Call for Papers

- **Tell us how you test with Jubula and get a goodie!**
Give Feedback on the Sessions

1. Sign In: www.eclipsecon.org

2. Select Session Evaluate

3. Vote

EVALUATE