




Experiences in agile testing

Hans-Joachim Brede
Bredex GmbH

© 2003 Bredex GmbH Version 2.4



AB 12.03.2008 2

Introduction

- Agile GUI testing challenges
- Agile GUI test design
- Project results and figures

BREDEX
Software-Entwicklung und Beratung

AB 12.03.2008 3

The challenges

- **Short release cycles**
Need continuous testing
- **Not much time**
Can't wait until delivery to start writing tests
Can't afford to spend too much time writing tests
Regression tests have to be stable
- **Requirements**
Flexible, subject to change

Test Automation

The chart 'Test Automation' shows the time spent on three activities over six iterations. The Y-axis is 'Time' and the X-axis is 'Iteration'. The legend indicates: Implementation (dark red), Test specification & maintenance (yellow), and Automatically executed tests (light blue).


Iteration	Implementation	Test specification & maintenance	Automatically executed tests
1	High	Medium	Low
2	High	Medium	Low
3	High	Medium	Low
4	High	Medium	Low
5	High	Medium	Low
6	High	Medium	Low

BREDEX
Software-Entwicklung und Beratung

AB 12.03.2008 4

Bad tests


- **Are dependent on availability of application**
- **Take too long/too many resources to write**
- **Have not been planned**



AB 12.03.2008 5

Good tests


- **Don't rely on application**
- **Easy and quick to create**
- **Need to be flexible**
- **Have been planned and are structured**



AB 12.03.2008 6

Designing agile GUI tests


- **Modular**
What actions/action combinations will be frequently used?
- **Abstract**
Separate from any implementations of application
Generic
Flexible
- **Keywords**



AB 12.03.2008 7

Project experience

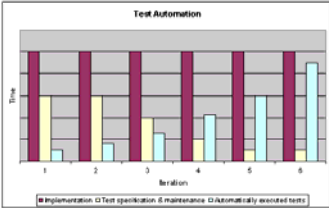
- Test and development in parallel**
- 10% of project costs
- On-time testing
- Prompt error discovery
- 0 errors on delivery



AB 12.03.2008 8

Summary

- To test GUI's agilely**
- Testing in synch with development
- Robust, well-thought out test design
- Tool choice**
- Levels of abstraction
- Not reliant on presence of application
- Supports quick & easy test creation/maintenance



Iteration	Implementation	Test specification	maintenance	Automatically executed tests
1	High	Low	Low	Low
2	High	Low	Low	Low
3	High	Low	Low	Low
4	High	Low	Low	Low
5	High	Low	Low	Low
6	High	Low	Low	Low