

# Selection and evaluation of GUI testing tool



© 2009 Tieto Corporation

Olga Smorodova

Tieto  
BA/BU/Industry  
[olga.smorodova@tieto.com](mailto:olga.smorodova@tieto.com)

**tieto**

# Table of contents

- Why do we need to automate our GUI testing?
- Expectations of the perfect GUI testing tool;
- GUI testing tools requirements;
- TestComplete (main possibilities, advantages and disadvantages);
- Reaching the goals;
- Statistics (TestComplete in Tieto, one automated TC example);
- Our next targets in test automation.



**Why do we need  
to automate our  
GUI testing?**

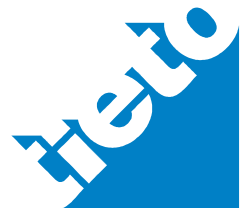
# Manual testing disadvantages:

- Each release, each platform multiplies the costs;
- Time consuming;
- Manual tests are most likely to be skipped when the project runs out of time;
- The tests are not reproducible, as testers tend to forget the actual steps they have taken until an error occurred;
- Manual testing between releases is impractical. So you have no early reports on possible problems.



# Expectations of the perfect GUI testing tool

- Capture and replay;
- Test Scripts saved as text;
- Testing language offers Try/Catch/Finally statements;
- GUI elements are recordable as mapping;
- Screenshots can be made;
- Test scripts can be modeled in a GUI;
- No new scripting language knowledge needed;
- And many more expectations such as quality of generated reports, support etc.



# GUI testing tool requirements

- XUL based application support;
- Record and play tool;
- Easy to follow test results (logs);
- No tool modifications needed (no open-source);
- No need to invest money in further tool development;
- Test automatic start;
- Easy to change incoming parameters;
- No difficulties to create a simple test case.



# TestComplete

Main possibilities, Advantages and Disadvantages of testing tool

*<http://www.automatedqa.com>*

# Main possibilities

- Windows and Web testing;
- Internet Explorer 8 and Firefox 3 support;
- Can perform functional testing, load testing, performance testing, client/server testing, integration testing, regression testing, data-driven testing;
- Record and Replay Automated Tests;
- Detailed Automated Testing Logs;
- Multiply scripting languages (VBScript, C#Script, C++Script, DelphiScript, JScript, VBScript);
- Etc.



# Advantages of TestComplete

- Records your steps;
- Multiple scripting language support;
- Easy to read log files;
- Easy to analyze reports, graphics;
- Checkpoints testing;
- Supports databases, Excel sheets and text files as external data sources;
- Tests can run automatically from command line.



# Disadvantages of TestComplete:

- Problems with some versions of Mozilla Firefox;
- Object recognition problems (depends on testing application difficulty);
- System requirements;
- Difficulties to understand cases written by another tester;
- Experience in testing automation.



# Reaching the goals

Benefits of using TestComplete in Tieto

- Errors and bugs can be found before patch/version is created;
- Main functionality is tested and it works correctly;
- Regression tests can run for each release;
- Resource and time-saving.



# Statistics

TestComplete in Tieto, one automated  
TC example

# TestComplete in Tieto

- TestComplete has been used in Tieto from August 2008;
- 4 application testing is automated;
- Tests runs 9 times a day;
- One application automated testing takes approx. 1 hour;
- 1 automated test case creation takes from 3h to 2days;
- One application automation takes 1-2 months.



# One automated TC example:

TC includes:

- New client, agreement, account and card generation;
- Client/agreement/account/  
Card, account search;
- Field checking;
- Interest calculator checking;
- Account, card data editing;
- Existing field changing;
- Data manipulation (puts card in stop-list);
- Saves data in file after new card/account creation and search them, saves data in another file, then compares 2 files.



# Our next targets in test automation

- Automate all GUI application testing;
- Load testing;
- Another language testing;
- Testing with all screen resolutions.



# Thank you

