

Outsource easily with us!

SOFTWARE TESTING PROCESSES
PRESENTATION

1. [Software testing \(ST\) tasks and objectives](#)
2. [ST documentation](#)
3. [Types of tests](#)
4. [Functional testing process](#)
5. [Process supporting tools](#)
6. [Performance testing](#)
7. [Automation testing](#)
8. [Localization testing](#)
9. [Security testing](#)
10. [Mobile testing](#)
11. [ScienceSoft ST department](#)

OUR GOAL IS EXCELLENCE. To reach this, we:

- Provide control over the product life-cycle
- Ensure the high quality of a product by controlling every stage of development
- Provide an accurate information about the product quality during the whole project life cycle

As a result, we produce quality software within defined scope and budget.

General documentation:

- Functional Testing process
- Integration Testing guidelines
- Unit Testing guidelines
- Defect Tracking guidelines
- Test Design Process

Document templates:

- Functional Specification
- Design Specification
- Test Plan
- Test Results Report

Specific documentation:

- Test Automation Process
- Document Review Process
- Requirements
- Specification guidelines
- Risk Management Guidelines
- File Management Policy
- Coding Standards

Functional testing – checking the features and operational behavior of a product to ensure they correspond to its specifications. This testing ignores the internal mechanism of a system or component and focuses solely on the outputs generated in response to selected inputs and execution conditions.

Performance testing – conducted to evaluate the compliance of a system or component with specified performance requirements. Often this is performed using an automated test tool to simulate large number of users.

Security testing – process to determine that an information system protects data and maintains functionality as intended. The six basic security concepts that need to be covered by security testing are: confidentiality, integrity, authentication, authorization, availability and non-repudiation.

Automation testing – the use of software to control the execution of tests, the comparison of actual outcomes to predicted outcomes, the setting up of test preconditions, and other test control and test reporting functions.

Localization testing - checking of translation; adaptation of interface elements, auxiliary files: documentation, help, IQ and Guide; correct justification and interface element accommodation, as well as provision for rules of text writing.

Usability testing is a technique used to evaluate a product by testing it on users. This can be seen as an irreplaceable usability practice, since it gives direct input on how real users use the system.

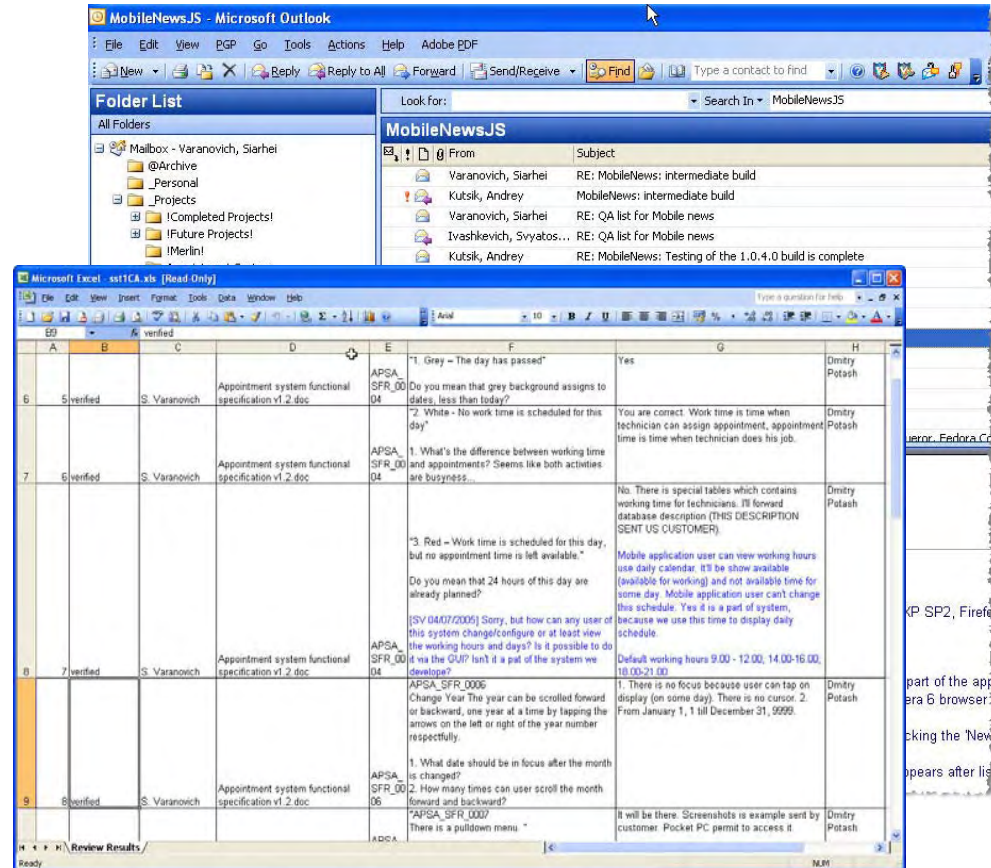


ACTIVITIES:

- Testers' assignment
- Kick-off meeting
- Study project related documentation, prototype and/or previous version of application

ARTIFACTS:

- Assignment letters
- The result of project analysis - Questions & Answers document

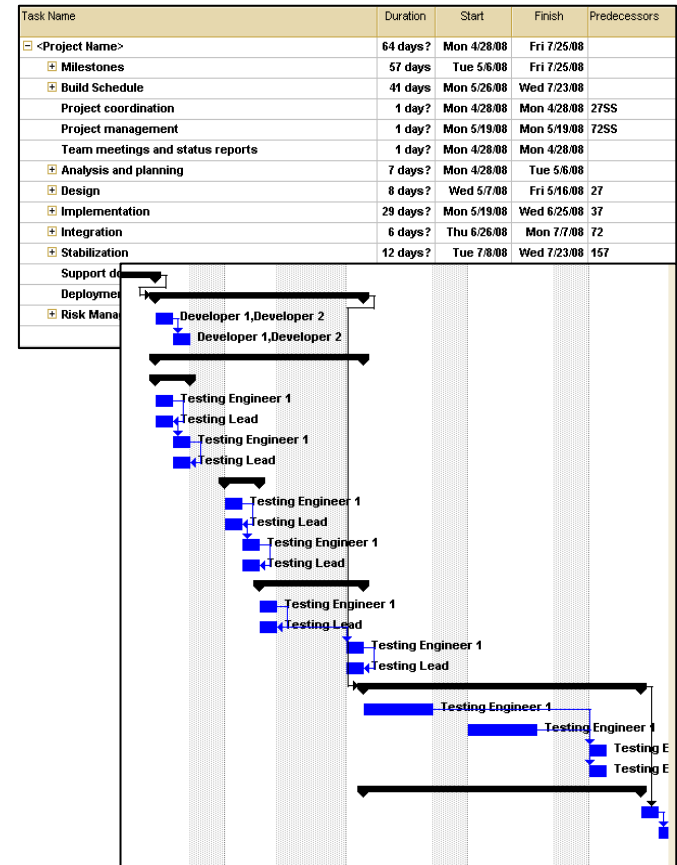


ACTIVITIES:

- Understand the operational (system, project and process) and the organizational context of the testing to be performed
- Define and prioritize the risks to system quality and obtain stakeholders consensus on the extent of testing to mitigate these risks
- Test Plan creation, approval and publishing
- Test environment establishment

ARTIFACTS:

- Test Plan
- Configuration Matrix (it may be included into Test Plan)
- Test hardware request (if needed)

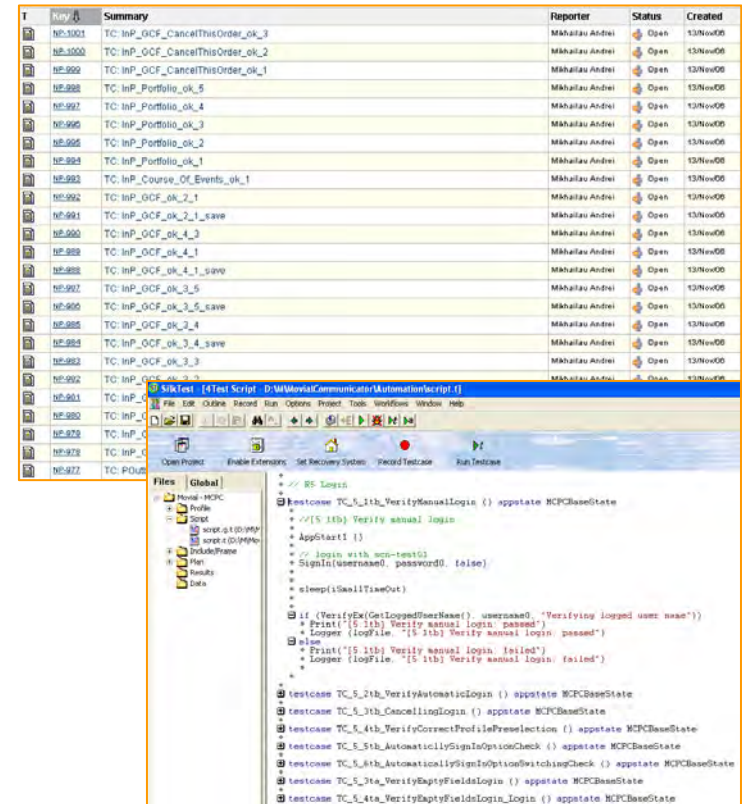


ACTIVITIES:

- Import functional specifications into the Jira (depends on the chosen approach)
- Test Procedures and Test Cases creation, confirmation and publishing
- Cover each functional requirement with one or more test cases
- Automated Test Scripts development (by customer's request)

ARTIFACTS:

- Test Cases in Jira
- Automated Test Scripts (by customer's request only)



The screenshot displays a test management interface. The top portion shows a table of test cases with columns for ID, Summary, Reporter, Status, and Created. The bottom portion shows a detailed view of a test script, including a file explorer on the left and a code editor on the right. The code editor contains a test script for verifying manual login, with comments and code blocks for login verification and automatic login.

ID	Summary	Reporter	Status	Created
NP-1001	TC: InP_GCF_CancelThisOrder_ok_3	Mikhailov Andrei	Open	13Nov06
NP-1000	TC: InP_GCF_CancelThisOrder_ok_2	Mikhailov Andrei	Open	13Nov06
NP-999	TC: InP_GCF_CancelThisOrder_ok_1	Mikhailov Andrei	Open	13Nov06
NP-998	TC: InP_Portfolio_ok_5	Mikhailov Andrei	Open	13Nov06
NP-997	TC: InP_Portfolio_ok_4	Mikhailov Andrei	Open	13Nov06
NP-996	TC: InP_Portfolio_ok_3	Mikhailov Andrei	Open	13Nov06
NP-995	TC: InP_Portfolio_ok_2	Mikhailov Andrei	Open	13Nov06
NP-994	TC: InP_Portfolio_ok_1	Mikhailov Andrei	Open	13Nov06
NP-993	TC: InP_Course_Of_Events_ok_1	Mikhailov Andrei	Open	13Nov06
NP-992	TC: InP_GCF_ok_2_1	Mikhailov Andrei	Open	13Nov06
NP-991	TC: InP_GCF_ok_2_1_save	Mikhailov Andrei	Open	13Nov06
NP-990	TC: InP_GCF_ok_4_3	Mikhailov Andrei	Open	13Nov06
NP-989	TC: InP_GCF_ok_4_1	Mikhailov Andrei	Open	13Nov06
NP-988	TC: InP_GCF_ok_4_1_save	Mikhailov Andrei	Open	13Nov06
NP-997	TC: InP_GCF_ok_3_5	Mikhailov Andrei	Open	13Nov06
NP-996	TC: InP_GCF_ok_3_5_save	Mikhailov Andrei	Open	13Nov06
NP-995	TC: InP_GCF_ok_3_4	Mikhailov Andrei	Open	13Nov06
NP-994	TC: InP_GCF_ok_3_4_save	Mikhailov Andrei	Open	13Nov06
NP-993	TC: InP_GCF_ok_3_3	Mikhailov Andrei	Open	13Nov06
NP-992	TC: InP_GCF_ok_3_2	Mikhailov Andrei	Open	13Nov06
NP-991	TC: InP_GCF_ok_3_1	Mikhailov Andrei	Open	13Nov06
NP-979	TC: InP_GCF_ok_3_1	Mikhailov Andrei	Open	13Nov06
NP-978	TC: InP_GCF_ok_3_1	Mikhailov Andrei	Open	13Nov06
NP-977	TC: InP_GCF_ok_3_1	Mikhailov Andrei	Open	13Nov06

```
// RS - Login
@Testcase TC_5_1tb_VerifyManualLogin () appstate MFCBaseState
//([5 1tb) Verify manual login
AppStart1 ()
// login with user=tests0
SignIn(user=tests0, password=, false)
sleep(10*60*1000)
if (Verify(Ex(GetLoggedInUserName), user=tests0, "Verifying logged user name"))
Print([5 1tb) Verify manual login: passed")
Logger(logFile, "[5 1tb) Verify manual login: passed")
else
Print([5 1tb) Verify manual login: failed")
Logger(logFile, "[5 1tb) Verify manual login: failed")

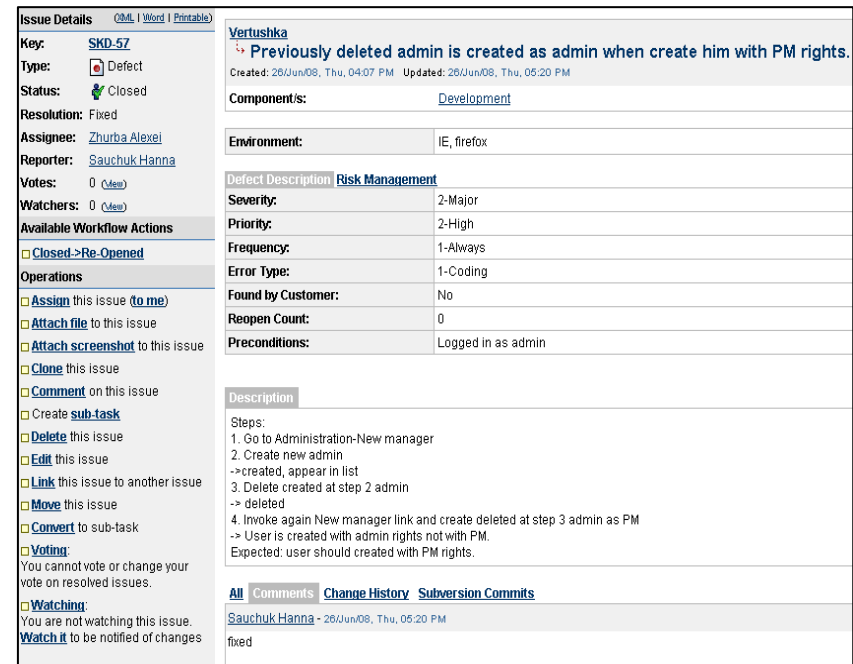
@Testcase TC_5_2tb_VerifyAutomaticLogin () appstate MFCBaseState
@Testcase TC_5_3tb_CancellingLogin () appstate MFCBaseState
@Testcase TC_5_4tb_VerifyCorrectProfilePreselection () appstate MFCBaseState
@Testcase TC_5_5tb_AutomaticallySignInWithOptionsCheck () appstate MFCBaseState
@Testcase TC_5_6tb_AutomaticallySignInWithOptionsSwitchingCheck () appstate MFCBaseState
@Testcase TC_5_3ta_VerifyEmptyFieldsLogin () appstate MFCBaseState
@Testcase TC_5_4ta_VerifyEmptyFieldsLogin_Login () appstate MFCBaseState
```

ACTIVITIES:

- Checking required configuration and obligatory documents (What'sNew, ReadMe etc.)
- Installation, Smoke Test execution. Executing all planned test cases accordingly to planned in TP strategies
- Defects reporting

ARTIFACTS:

- Notification – planned test suit passed/failed
- Defect Reports



The screenshot shows a defect report interface. On the left, there are tabs for 'Issue Details', 'Comments', 'Word', and 'Printable'. The 'Issue Details' tab is active, showing the following information:

- Key:** SKD-57
- Type:** Defect
- Status:** Closed
- Resolution:** Fixed
- Assignee:** Zhurba Alexei
- Reporter:** Sauchuk Hanna
- Votes:** 0
- Watchers:** 0

Below this information is a section for 'Available Workflow Actions' with a link for 'Closed->Re-Opened'. Underneath is an 'Operations' section with various actions like 'Assign this issue (to me)', 'Attach file to this issue', 'Attach screenshot to this issue', 'Clone this issue', 'Comment on this issue', 'Create sub-task', 'Delete this issue', 'Edit this issue', 'Link this issue to another issue', 'Move this issue', 'Convert to sub-task', 'Voting', and 'Watching'.

On the right side of the interface, the defect details are shown:

- Defect Title:** Previously deleted admin is created as admin when create him with PM rights.
- Created:** 26/Jun/08, Thu, 04:07 PM
- Updated:** 26/Jun/08, Thu, 05:20 PM
- Component:** Development
- Environment:** IE, firefox
- Defect Description:** Risk Management
- Severity:** 2-Major
- Priority:** 2-High
- Frequency:** 1-Always
- Error Type:** 1-Coding
- Found by Customer:** No
- Reopen Count:** 0
- Preconditions:** Logged in as admin

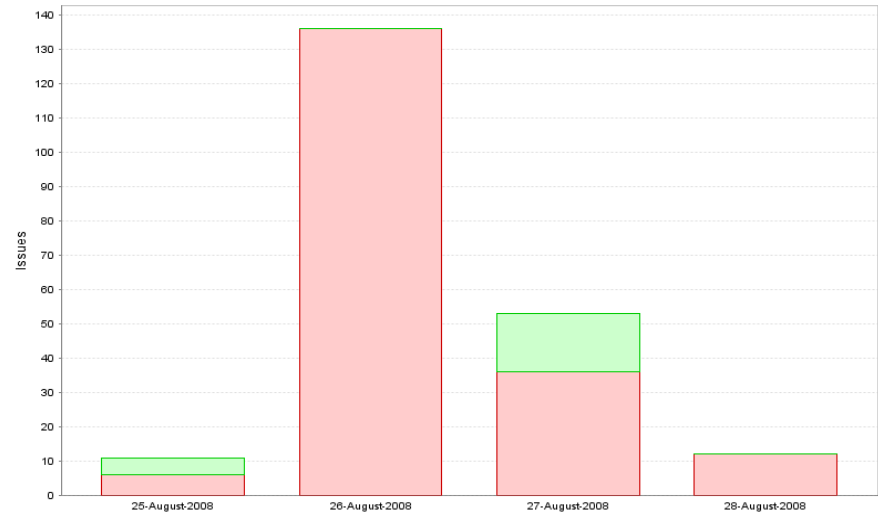
Below the defect details is a 'Description' section with the following steps:

1. Go to Administration-New manager
2. Create new admin
-> created, appear in list
3. Delete created at step 2 admin
-> deleted
4. Invoke again New manager link and create deleted at step 3 admin as PM
-> User is created with admin rights not with PM.
Expected: user should created with PM rights.

At the bottom, there are tabs for 'All', 'Comments', 'Change History', and 'Subversion Commits'. The 'Comments' tab is active, showing a comment from 'Sauchuk Hanna' dated '26/Jun/08, Thu, 05:20 PM' with the text 'fixed'.

ACTIVITIES:

- Defect metrics accounting and analyzing
- Communicating test results to key stakeholders. Weekly reporting



ARTIFACTS:

- Test Results Report

Period	Created Issues (Unresolved)	Created Issues (Resolved)
25-August-2008	6	5
26-August-2008	136	0
27-August-2008	36	17
28-August-2008	12	0

ACTIVITIES:

- Final build acceptance and recommendation for production
- Overall testing results summarization and evaluation
- Participation in the Postmortem meeting

ARTIFACTS:

- Test Evaluation Summary report
- Post-Mortem Report, testing related sections

Process	Tool
Project Planning	MS Project
Test and Defect tracking	Atlassian Jira* , LogiGear TrackGear, HP Quality Center, Bugzilla and Rational ClearQuest
Configuration management	SVN, MS VSS, CVS and SharePoint
Process monitoring	Atlassian Jira* , LogiGear TrackGear, HP Quality Center, Rational ClearQuest and system of metrics& reports
Test automation	Borland SilkTest, Rational Functional Tester, Selenium, Borland SilkPerformer, HP LoadRunner, HP WinRunner, HP QTP, HttpUnit, JMeter and Rational Performance Tester etc.
Security test tools	Wireshark, Watchfire Appscan, WebInspect, Positive Technologies XSpider, small tools for injection checks
Risk management	Report system, MS Project

* - we have customization Atlassian Jira.



Tools used for performance testing:

- Rational Performance tester
- HP LoadRunner
- Borland SilkPerformer
- JMeter

- Measure end-to-end response time for client/server environment
- Measure hardware resource usage on all application and database servers, as well as any additional devices that take part in the system environment (network, load balancer etc.)
- Verify whether service level requirements are met
- Identify possible bottlenecks in the system
- Provide analysis of data to define a configuration that will offer optimum performance
- Verify reliability of the system
- Define possible breakpoints, determine the maximum load the server can handle before degradation in response time occurs and propose changes to optimize performance of an application.

All scripts' results can be introduced in lots of graphic reports.

Example of script file:

```
Project 'Wicket'
├── Profiles
├── Scripts (4)
├── Data Files
├── Agents
└── Workloads

benchmark SilkPerformerRecorder

//Include files
use "WebAPI.bdh"
use "Kernel.bdh"

dcluser
user
VUser
transactions
TInit      : begin;
//Count    : 1;
TMain      : 1;

var
//login: string;

hFile : number;
sColumn : string;
sColumn2, sLink, sUrl, sLink2, sUrl2 : string;

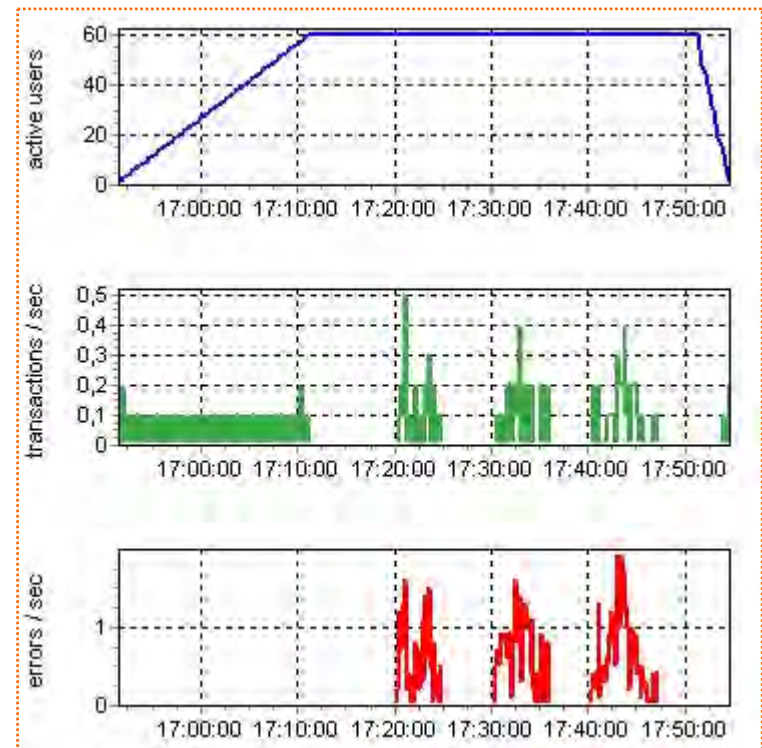
dclrand
r : RndPerN(238);
gname : RndFile("gname.rnd", 50);
LDAP : RndFile("LDAP.rnd", 40);
id : RndPerN(218);
gname2 : RndFile("gname2.rnd", 50);

//rLoginTime: RndUniF(0.0..3.0);

dcltrans
transaction TInit
begin

WebSetBrowser(WEB_BROWSER_NS40);
```

Example of generated graphics in report:



Benefits of using automated testing:

1. Automated testing saves time and money.
2. Automated testing minimizes the effort involved in manual testing.
3. Automated testing increases software quality.
4. Automated testing can be run over and over again with less overhead.
5. Automated testing reduces time to-market.

Test automation is:

- the use of software to control the execution of tests
- the comparison of actual outcomes to predicted outcomes
- the setting up of test preconditions
- other test control and test reporting functions.

Test automation involves automating a manual process already in place that uses a formalized testing process. It's usually executed in addition to manual testing to provide more qualitative software products.

It can be made in the longer term though, especially in:

- Regression Testing
- Data Driven Testing (the same scenario & different input data)
- Performance & Load Testing

Unit Testing Tools:

xUnit (CPPUnit, JUnit etc)

Middleware Tools:

HTTPUnit

GUI Tools:

Borland Silk Test, IBM Rational Functional Tester, HP WinRunner and HP QTP

Performance Tools:

JMeter, Selenium, Borland SilkPerformer, HP LoadRunner and Rational Performance Tester etc.

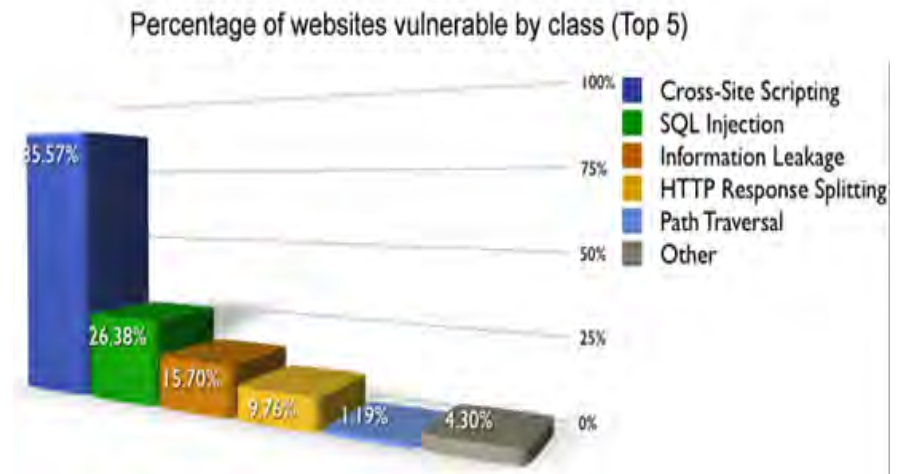
- 80% of ST staff is experienced in Borland SilkTest, HP WinRunner or QuickTestPro, Borland Silk Performer and Rational products.
- ST team have advanced test automation skills and experience (regression, new features testing and service tools creation).
- Experience in different approaches to automation: development scripts, framework creation and data-driven testing.
- We are engaged in a big projects scripts' development (client/server applications, database products, educational software, embedded software, standalone and web applications).
- ST team organize and take part in self assessment and automation trainings.

- Localization Testing process includes the testing of the product that was adapted for use in a foreign market.
- Successful localization products contain high quality native language translation and retains the functionality of the product.
- Localization must also consider how various cultural conventions in the target country require implementing source code changes.

- Products that are localized to international markets often face domestic competition, which makes it critical for the localized product to harmonize easy into the native language and cultural landscape
- The cost of a localization effort can be significant. Once we have the strings translated and the GUI updated, localization testing should be used to help ensure that the product is successfully migrated to the target market. In addition to verifying successful translation, basic functional testing should be performed
- Functional issues often arise as a result of localizing software
- Do not risk the time and effort spent localizing by not performing corresponding Software Testing

Benefits of security tests execution:

1. Security testing helps to improve application quality and minimize risks from network hooligans
2. Security testing saves us from losing clients, disappointed by privacy lack or important information loose
3. Meeting market security requirements we can get a market advantage for our application



Security testing - a process to determine that an information system protects data and maintains functionality as intended.

The six basic security concepts that need to be covered by security testing are:

- Integrity
- Confidentiality
- Authentication
- Authorization
- Availability
- Non-Repudiation

1. Growing market.

In 2008 alone there'll be an estimated 58 million PDAs sold worldwide (source: [eTForecast](#))

2. Less users experience

People generally have a lot less experience of using their mobile to go online than they do of using their computer.

3. Mobile platforms variety

The platform through which users access site is far less predictable when using mobile phones.

Mobile applications testing - functional testing of mobile-based applications.

Often no special tools used, but a large set of devices is required. There are two ways to organize testing:

- **Using a set of devices.**

Different devices needs to be checked, as every device have own bugs and differences in system features implementation.

- **Using software device emulators.**

This way is not so reliable, as device specific can't be emulated, but can be used for standardized lines of devices.

Programming: C/C++/C#, Java, Object Pascal, Visual Basic, JavaScript, VBScript, HTML, .NET, PHP and Perl

Database Administration: Oracle, MS SQL, IBM DB2, Sybase, Informix and MySQL

System Administration: Windows NT, Windows 2000, Windows 2003, Sun Solaris, HP-UX, IBM AIX, Linux and OS 390/400

Network Administration: NetWare, NT, Cisco IOS, TCP/IP, IPX/SPX and NetBIOS

Web Servers Administration: IIS, Apache and IBM HTTP

Application Servers Administration: WebSphere, WebLogic, JRun and Tomcat

Automated Testing: Borland SilkTest and SilkPerformer, HP Interactive, WinRunner, LoadRunner and QuickTestPro, Rational XDE Tester, HTTP Unit, JMeter and VS Team Edition.

- Our development processes and quality standards ensure that the delivered product will be of high quality or at least the quality is controlled
- Our system of reports guarantees that all the project stakeholders are always informed about the quality of the product in each particular moment and have enough information to make decisions
- Our processes are flexible, we can customize them for needs of any customer and can adopt almost any tool or practice that is suitable for our customer
- Employees in the test team are not focused in some particular area, have knowledge in broad variety of domains and are easy learners. As a result, minimal time would be required for members of our test team to start working efficiently on your project

Thank you

YOUR QUESTIONS ARE WELCOME !