

**TEST DESIGN SPECIFICATION**

DEMO APPLICATION

UI ◦ SERVICE ◦ PERFORMANCE   
TEST CASES

Vasilis Petrou ◦ 2019

Contents

[1 Introduction 5](#_Toc27007310)

[1.1 Purpose 5](#_Toc27007311)

[1.2 Scope 5](#_Toc27007312)

[1.3 Target Audience 5](#_Toc27007313)

[1.4 Structure of this document 5](#_Toc27007314)

[1.5 Reference and applicable documents 6](#_Toc27007315)

[1.5.1 Reference Documents 6](#_Toc27007316)

[1.5.2 Applicable Documents 6](#_Toc27007317)

[1.6 Abbreviations and Acronyms 6](#_Toc27007318)

[1.7 Definitions 7](#_Toc27007319)

[2 Approach Refinements 8](#_Toc27007320)

[2.1 UI Test Cases Package 8](#_Toc27007321)

[2.2 WS Test Cases Package 8](#_Toc27007322)

[2.3 Performance Test Cases Package 9](#_Toc27007323)

[3 Environments Description 10](#_Toc27007324)

[3.1 Software Pre-Requisites 10](#_Toc27007325)

[3.2 UI Test Execution & Reporting 10](#_Toc27007326)

[3.2.1 Installation of the UI Test package 10](#_Toc27007327)

[3.2.2 Execution of UI Test Case/Suite 11](#_Toc27007328)

[3.2.3 Reporting of UI Test Case/Suite 11](#_Toc27007329)

[3.3 WS Test Execution & Reporting 12](#_Toc27007330)

[3.3.1 Execution of WS Test Case/Suite 12](#_Toc27007331)

[3.3.2 Reporting of WS Test Case/Suite 12](#_Toc27007332)

[3.3.3 Execution of CleanupDB Test Case 12](#_Toc27007333)

[3.3.4 Reporting of CleanupDB Test Case 12](#_Toc27007334)

[3.4 Performance Test Execution & Reporting 13](#_Toc27007335)

[3.4.1 Execution of Performance Test Case/Suite 13](#_Toc27007336)

[3.4.2 Reporting of WS Test Case/Suite 13](#_Toc27007337)

[4 Test Cases 14](#_Toc27007338)

[4.1 UI Test Cases 14](#_Toc27007339)

[5 Traceability to Test Suites 15](#_Toc27007340)

# Introduction

## Purpose

This document describes the Test Cases that will be followed in order to test the DEMO application from the User Interface, Web Service and Performance point of view. The document will be used to guide the execution of all Test Cases defined as a means of validating the implementation against the DEMO Graphical User Interface, Web Service and Performance Specifications. Information on detailing Test Cases with execution steps, the expected results as well as the test pass criteria are also included.

The purpose of the Test Cases is to provide guidelines for validating the DEMO Application implemented web pages. The functionality and business logic behind the user interface are covered by the service testing. Nonfunctional requirements are covered by the performance testing.

## Scope

This document is a template to be shared for training reasons.

## Target Audience

N/A

## Structure of this document

The present document contains the following chapters:

* **Chapter 1 - Introduction:** describes the scope and the objectives of the document;
* **Chapter** **2 - Approach Refinements:** describes the test case packages;
* **Chapter 3 - Environments Description:** lists the requirements of the testing environment and provides details on the execution of UI Test Cases/Suites;
* **Chapter 4 – Test Cases:** describes the actual UI/Service/Performance Test Cases;
* **Chapter 5 - Traceability to Test Suites:** provides a traceability matrix to Test Suites.

## Reference and applicable documents

### Reference Documents

| Ref. | Title | Reference | Version | Date | |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | |  |

Table 1: Reference Documents

### Applicable Documents

| Ref. | Title | Reference | Version | Date | |
| --- | --- | --- | --- | --- | --- |
|  |  |  |  | |  |

Table 2: Applicable Documents

## Abbreviations and Acronyms

For a better understanding of the present document, the following table provides a list of the principal abbreviations and acronyms used.

| **Abbreviation/Acronym** | **Definition** |
| --- | --- |
| ATT | Automated Testing Tool |
| DB | Database |
| GUI | Graphical User Interface |
| ID | Identification number |
| IT | Information Technology |
| SDLC | Software Development LifeCycle |
| TC | Test Case |
| TDS | Test Design Specification |
| UI | User Interface |
| WS | Web Service |

Table 3: Abbreviations and Acronyms

## Definitions

| **Name** | **Description** |
| --- | --- |
| Test Case | A test case is a set of steps applied to transit the system from a known initial state to an expected final state. It is characterised by:   * An identifier that uniquely identifies the test case; * A short description of its purpose; * One or several actors performing the actions on the system; * A set of pre-requisites describing the initial state of the system; * A set of steps defining the actions performed by the actors and their expected results. Steps may describe input values and expected output values if applicable for the tested feature; * A set of post-conditions describing the state in which the system is expected to be after the execution of the steps; * A set of pass criteria defining the conditions under which the test can be considered as passed.   A test scenario is realised by one or several test cases without an explicit mapping between the test cases and the test scenario guidelines. |
| Test Suite | A Test Suite is a set of Test Cases for a component/service or system under test. It is usually used to group and/or facilitate the execution a number of similar Test Cases. |

Table 4: Definitions

# Approach Refinements

## UI Test Cases Package

The Test Package contains the following folder:

* GuiTests

The UI Test Cases Package (folder GuiTests) contains the following sub-folders/files:

| **File / Folder** | **Description** |
| --- | --- |
| **src** | Folder that contains the UITT Source Code |
| **pom.xml** | File that is needed for the installation with maven. |
| **.gitignore** | A gitignore file specifies intentionally untracked files that Git should ignore. |

Table 5: Contents of the UI Test Cases Package

## WS Test Cases Package

The Test Package contains the following folder:

* SrvTests

The Web Services Test Cases Package (folder SrvTests) contains the following sub-folders/files:

| **File / Folder** | **Description** |
| --- | --- |
| **src** | Folder that contains the SoapUI Project and Properties file with global variables. |
| **pom.xml** | File that is needed for the installation with maven. |
| **.gitignore** | A gitignore file specifies intentionally untracked files that Git should ignore. |
| **launch-test.sh** | A shell script that is needed for the full execution of WS tests and the generation of a report. |
| **cleanup.sh** | A shell script that is needed for the cleanup of data and generation of a report. |

Table 6: Contents of the WS Test Cases Package

## Performance Test Cases Package

The Test Package contains the following folder:

* PerfTests

The Performance Test Cases Package (folder PerfTests) contains the following sub-folders/files:

| **File / Folder** | **Description** |
| --- | --- |
| **src** | Folder that contains the JMeter Project and Recources that are needed for the reporting. |
| **pom.xml** | File that is needed for the installation with maven. |
| **.gitignore** | A gitignore file specifies intentionally untracked files that Git should ignore. |

Table 7: Contents of the UI Test Cases Package

# Environments Description

The following sub-sections contain details on:

* the software pre-requisites for the UI Test Cases Package;
* the instructions on how to execute a UI Test Case/Suite;
* the report(s) generated automatically after the execution of a UI Test Case/Suite.

## Software Pre-Requisites

* The DEMO Application must be correctly installed and configured on the appropriate server;
* The following software must be available on the machine from where the UI Test Cases/Suites will be executed:
  + Java JDK 1.8.0[[1]](#footnote-1);

*dependency required for the execution of Test Cases/Suites;*

* + Google Chrome

*dependency required for the execution of Test Cases/Suites*

* + GIT for Windows[[2]](#footnote-2) ;

*dependency required for the execution of the scripts;*

* + Apache Maven[[3]](#footnote-3);

*dependency required for the installation of the test packages and execution of Test Cases/Suites;*

* + SoapUI Open Source[[4]](#footnote-4);

*dependency required for the processing of the WS test scripts;*

* + Apache JMeter[[5]](#footnote-5);

*dependency required for the processing of the Performance test scripts;*

## UI Test Execution & Reporting

### Installation of the UI Test package

Assuming that the UI Test Cases Package was extracted to “*C:\DEMO\GuiTests”*, open Git Bash and execute the following commands:

|  |
| --- |
| *cd /c/DEMO/GuiTests*  *mvn clean install* |

After successful installation, the UITT is generated in *C:\DEMO\GuiTests\target*

The contents of the UITT Test Tool are presented below:

| **File / Folder** | **Description** |
| --- | --- |
| **test\_data** | Sub-folder that contains clean-up scripts and a list of files needed for upload/load operations. |
| **test\_suites** | Sub-folder that contains all UI Test Suite XML files. |
| **chromedriver.exe** | A separate executable that Selenium WebDriver uses to control Chrome. |
| **demo-test-1.0.jar** | Executable jar file that is needed for UI Test Case/Suite execution. |
| **launch-test.sh** | Main shell script that is needed for the UI Test Case/Suite execution. |

Table 8: Contents of the UI Test Cases Tool

### Execution of UI Test Case/Suite

Assuming the UI Test Tool was installed in the *C:\DEMO\GuiTests\target*

• Open Git Bash;

• Navigate to the UI Test Tool folder’s path (e.g. *cd C:\DEMO\GuiTests\target*).

To execute *all test suites*, the following command can be executed:

|  |
| --- |
| *./launch-test.sh* |

### Reporting of UI Test Case/Suite

After execution of a UI Test Case/Suite, a report is generated automatically in a folder named by:

***\reports\<name of test case/suite>\_<timestamp>***

Within that folder and within sub-folder “html”, an “index.html” file will reside with the results of the report for the UI Test Case/Suite that was executed.

## WS Test Execution & Reporting

### Execution of WS Test Case/Suite

Assuming that the WS Test Cases Package was extracted to “*C:\DEMO\SrvTests”*, open Git Bash and execute the following commands:

|  |
| --- |
| *cd /c/DEMO/SrvTests*  *./launch-test.sh* |

### Reporting of WS Test Case/Suite

After execution of a WS Test Case/Suite, a report is generated automatically in a folder named by:

***\test-reports\DemoAppWS\_TestReport\_<timestamp>***

Within that folder, a “surefire-report.html” file will reside with the results of the report for the WS Test Case/Suite that was executed.

### Execution of CleanupDB Test Case

Assuming that the WS Test Cases Package was extracted to “*C:\DEMO\SrvTests”*, open Git Bash and execute the following commands:

|  |
| --- |
| *cd /c/DEMO/SrvTests*  *./cleanup.sh* |

### Reporting of CleanupDB Test Case

After execution of the CleanupDB WS Test Case, a report is generated automatically in a folder named by:

***\test-reports\DemoAppCleanupDB\_TestReport\_<timestamp>***

Within that folder, a “surefire-report.html” file will reside with the results of the report for the CleanupDB WS Test Case that was executed.

## Performance Test Execution & Reporting

### Execution of Performance Test Case/Suite

Assuming that the Performance Test Cases Package was extracted to “*C:\DEMO\PerfTests”*, open Git Bash and execute the following commands:

|  |
| --- |
| *cd /c/DEMO/PerfTests*  *mvn clean verify*  *mvn pre-site* |

### Reporting of WS Test Case/Suite

After execution of a WS Test Case/Suite, a report is generated automatically in a folder named by:

***\target\jmeter\results\dashboard***

Within that folder, a “index.html” file will reside with the results of the report for the WS Test Case/Suite that was executed.

*NOTE: If we get an exception* ***java.net.BindException: Address already in use: connect*** *this issue has to do with the Windows OS and registry must be updated as described below:*

* *Start Registry Editor (regedit)*
* *Locate the following subkey in the registry, and then click Parameters: HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters*
* *Right mouse clicked on Parameters.*
* *Create a new DWORD value with name MaxUserPort*
* *Right mouse click on MaxUserPort .*
* *then enter 65534 for the Value Data field.*
* *Under the radio button selection for Base, select the Decimal radio button.*
* *Restart the machine.*

# Test Cases

## UI Test Cases

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | | *TC\_CH\_UI\_001\_CreateNewContact* | | | |
| Purpose | | *Test the creation of a Contact inserting all the data.* | | | |
| Pre-Conditions | | *The DEMO Application is up and running.  List of Contacts is cleaned-up.* | | | |
| Basic Flow of Events | | | | | |
| Actor(s) Actions | | | Data | System Response | |
| 1 | *Open DEMO Application.* | |  | 1 | *The “Contact List” page opens.* |
| 2 | *Click on “Add New Contact” button from the menu.* | |  | 2 | *The “Add New Contact” page opens.* |
| 3 | *Enter the following “Contact” data.* | | * *Name=Contact 001* * *Address=Address 001* * *City=Athens* * *Phone=2101111111* * *Email=email001@intrasoft-intl.com* * *Status=enabled* * *Gender=male* | 3 | *The “Contact” data are inserted in the corresponding fields.* |
| 4 | *Click on “Save” button.* | |  | 4 | *After successful Save action, it redirects to the “Contact List” page* |
| 5 | *Verify that new Contact exists in the list of contacts.* | | * *Email=email001@intrasoft-intl.com* | 5 | *The new Contact exists in the list of contacts.* |
| Pass Criteria | | *The contact is created successfully and is displayed in the list of contacts.* | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | | *TC\_CH\_UI\_002\_CreateNewContactOnlyMandatory* | | | |
| Purpose | | *Test the creation of a Contact inserting only the mandatory data.* | | | |
| Pre-Conditions | | *The DEMO Application is up and running.  List of Contacts is cleaned-up.* | | | |
| Basic Flow of Events | | | | | |
| Actor(s) Actions | | | Data | System Response | |
| 1 | *Open DEMO Application.* | |  | 1 | *The “Contact List” page opens.* |
| 2 | *Click on “Add New Contact” button from the menu.* | |  | 2 | *The “Add New Contact” page opens.* |
| 3 | *Enter the following “Contact” data.* | | * *Name=Contact 002* * *City=Thessaloniki* * *Email=email002@intrasoft-intl.com* | 3 | *The “Contact” data are inserted in the corresponding fields.* |
| 4 | *Click on “Save” button.* | |  | 4 | *After successful Save action, it redirects to the “Contact List” page* |
| 5 | *Verify that new Contact exists in the list of contacts.* | | * *Email=email002@intrasoft-intl.com* | 5 | *The new Contact exists in the list of contacts.* |
| Pass Criteria | | *The contact is created successfully and is displayed in the list of contacts.* | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | | *TC\_CH\_UI\_003\_SearchAndViewContact* | | | |
| Purpose | | *Test the searching and view of a Contact.* | | | |
| Pre-Conditions | | *The DEMO Application is up and running.  List of Contacts is cleaned-up.*  *TC\_CH\_UI\_001\_CreateNewContact was executed successfully and contact with name Contact 001 exists.* | | | |
| Basic Flow of Events | | | | | |
| Actor(s) Actions | | | Data | System Response | |
| 1 | *Open DEMO Application.* | |  | 1 | *The “Contact List” page opens.* |
| 2 | *Search for a specific contact* | | * *Filter=Contact 001* | 2 | *The contact exists in the List of Contacts.* |
| 3 | *Click on “Details” button* | |  | 3 | *The “View Contact” page opens with all data in view mode and 3 buttons Edit/Back/Delete* |
| 4 | *Click on “Back” button* | |  | 4 | *The “Contact List” page opens.* |
| Pass Criteria | | *The contact is found and viewed successfully. Button “Back” from view page works properly.* | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | | *TC\_CH\_UI\_004\_SearchAndEditContact* | | | |
| Purpose | | *Test the searching and update of a Contact.* | | | |
| Pre-Conditions | | *The DEMO Application is up and running.  List of Contacts is cleaned-up.*  *TC\_CH\_UI\_001\_CreateNewContact was executed successfully and contact with name Contact 001 exists.* | | | |
| Basic Flow of Events | | | | | |
| Actor(s) Actions | | | Data | System Response | |
| 1 | *Open DEMO Application.* | |  | 1 | *The “Contact List” page opens.* |
| 2 | *Search for a specific contact* | | * *Filter=Contact 001* | 2 | *The contact exists in the List of Contacts.* |
| 3 | *Click on “Details” button* | |  | 3 | *The “View Contact” page opens with all data in view mode and 3 buttons Edit/Back/Delete* |
| 4 | *Click on “Edit” button* | |  | 4 | *The “Edit Contact” page opens.* |
| 5 | *Update the following “Contact” data.* | | * *Name=Contact 001a* * *Address=Address 001a*   *Email=email001a@intrasoft-intl.com* | 5 | *The “Contact” data are inserted in the corresponding fields.* |
| 6 | *Click on “Update” button* | |  | 6 | *The “View Contact” page opens with all data in view mode and with updated values.* |
| Pass Criteria | | *The contact is found and updated successfully. Button “Edit” from view page and button “Update” from the edit page, work properly.* | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ID | | *TC\_CH\_UI\_005\_DeleteContacts* | | | |
| Purpose | | *Test the deletion of a Contact.* | | | |
| Pre-Conditions | | *The DEMO Application is up and running.  List of Contacts is cleaned-up.*  *TC\_CH\_UI\_001\_CreateNewContact and TC\_CH\_UI\_002\_CreateNewContactOnlyMandatory were executed successfully and contact with name Contact 001 exists.* | | | |
| Basic Flow of Events | | | | | |
| Actor(s) Actions | | | Data | System Response | |
| 1 | *Open DEMO Application.* | |  | 1 | *The “Contact List” page opens.* |
| 2 | *Search for a specific contact* | | * *Filter=Contact 001* | 2 | *The contact exists in the List of Contacts.* |
| 3 | *Click on “Details” button* | |  | 3 | *The “View Contact” page opens with all data in view mode and 3 buttons Edit/Back/Delete* |
| 4 | *Click on “Delete” button* | |  | 4 | *A popup window opens so the user to confirm deletion.* |
| 5 | *Click on “OK” button of the confirmation popup window* | |  | 5 | *The “Contact List” page opens.* |
| 6 | *Search for a specific contact* | | * *Filter=Contact 002* | 6 | *The contact exists in the List of Contacts.* |
| 7 | *Click on “Details” button* | |  | 7 | *The “View Contact” page opens with all data in view mode and 3 buttons Edit/Back/Delete* |
| 8 | *Click on “Delete” button* | |  | 8 | *A popup window opens so the user to confirm deletion.* |
| 9 | *Click on “OK” button of the confirmation popup window* | |  | 9 | *The “Contact List” page opens.* |
| 10 | *Verify that Contact List is empty.* | |  |  | *The List of Contacts contains no record.* |
| Pass Criteria | | *The contact is found and deleted successfully. After deletion, list of contacts is updated properly.* | | | |

## WS Test Cases

|  |  |  |  |
| --- | --- | --- | --- |
| Service Name | Service Operation | Test Case ID | Test Case Description |
| ContactsREST | Create | TC\_CH\_SRV\_01 | 1 TC where a contact with all the fields is created successfully. |
| ContactsREST | View | TC\_CH\_SRV\_02 | 1 TC where a contact is viewed successfully. |
| ContactsREST | Update | TC\_CH\_SRV\_03 | 1 TC where a contact is updated successfully. |
| ContactsREST | Delete | TC\_CH\_SRV\_04 | 1 TC where a contact is deleted successfully. |
| ContactsREST | GetAllContacts | TC\_CH\_SRV\_05 | 1 TC where a list of all contacts is retrieved successfully. |

## Performance Test Cases

|  |  |  |  |
| --- | --- | --- | --- |
| Service Name | Service Operation | Test Case ID | Test Case Description |
| ContactsREST | GetAllContacts | TC\_CH\_PERF\_01 | 1 TC where a list of all contacts is retrieved successfully.  100 users that reach gradually the system in 10 seconds, request this list of contacts for 30 seconds.  Number of Threads (users): 100  Ramp-up period (seconds): 10  Duration (seconds): 30  Pass Criteria: Average Response Time < 1 second (with at least 30 existing contacts) |

# Traceability to Test Suites

The traceability between the Test Cases and the Test Suites is defined in the following table:

| Test Suite ID | Test Case ID | No of TCs |
| --- | --- | --- |
| TS\_CH\_UI | TC\_CH\_UI\_001\_CreateNewContact | 5 |
| TC\_CH\_UI\_002\_CreateNewContactOnlyMandatory |
| TC\_CH\_UI\_003\_SearchAndViewContact |
| TC\_CH\_UI\_004\_SearchAndEditContact |
| TC\_CH\_UI\_005\_DeleteContacts |
| TS\_CH\_SRV | TC\_CH\_SRV\_01 | 5 |
| TC\_CH\_SRV\_02 |
| TC\_CH\_SRV\_03 |
| TC\_CH\_SRV\_04 |
| TC\_CH\_SRV\_05 |
| TS\_CH\_PERF | TC\_CH\_PERF\_01 | 1 |

*\*\*\* End of document \*\*\**

1. <https://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html> [↑](#footnote-ref-1)
2. <https://git-scm.com/download/win> [↑](#footnote-ref-2)
3. https://maven.apache.org/download.cgi [↑](#footnote-ref-3)
4. https://www.soapui.org/downloads/soapui.html [↑](#footnote-ref-4)
5. https://jmeter.apache.org/download\_jmeter.cgi [↑](#footnote-ref-5)