

The aim of the SVVT process is to ensure that a new or changed service will provide the expected value, utility and warranty, to the business.

The benefits of testing properly services before being deployed into the production environments are basically the following:

- Customer's satisfaction increase, since the services are delivering the desired value to the users.
- Less requests to the Helpdesk for support, since services are operating as expected.
- Less incidents, since the functionalities and performance have been tested and match with the requirements.

In the context of this procedures software developers may be requested, in line with their contractual obligations to provide support in the development of the test Plan. The Annex 4 of the procedure shows examples of the content of a test plan (Template I) and test case specification (Template II).

The Test Plan constitutes the guideline by which the application will be tested, defining the acceptance criteria and the tests to be executed. In general the Test Plan makes reference to functional and non-functional tests under the following categories:

- Smoke tests: to check out if the main function of an application work properly, but not going into great details. They are put in place before the exhaustive testing activities or after the deployment on a different environment.
- Functional tests: to check out the functional behaviour of the application, including regression, in respect to specific function/feature.
- Non-Functional test: security, integration, load, stress, soak, and Business Continuity Facility (BCF) compliance. Load, stress and soak test descriptions must define all test scenarios to be executed, goals to be achieved, tools to be used and resources to be monitored.
- Acceptance (Business Validation) test: to validate that the business requirements are fulfilled by the new release. The 'Acceptance' formally acknowledges that the release has met the requirements once the release is deployed into the production environment. The 'Acceptance Criteria' should be defined by the business representatives and include at least a prioritized list of defined and measurable attributes, which are application-related, that must be satisfied to achieve the final acceptance.

The test procedures must aim at ensuring the compliance and quality of the release and to cause minimum disruption to the already in-place production environment. Furthermore should aim in verifying that the binaries resulted from the software built in-house are correct, can be used for installing the application in the production environment and once installed achieve the desired results.

All tests description must define all test scenarios to be executed, goals to be achieved, tools to be used, and resources to be monitored, when applicable<sup>1</sup>.

Within the context of site acceptance activities a relevant step that is performed is Software Quality Assurance<sup>2</sup>. Software Quality Assurance aims to:

- Assess and assure **Source Code Suitability** (Reliability, Correctness, Accuracy, Efficiency, and Usability) to identify issues/potential problems, propose solutions/changes and confirm implementation results.

---

<sup>1</sup> TestLink shall be used as platform to support the different testing activities, as in the case of the RDM procedure, the product(s) (i.e. Test Plan) shall be stored in TeamForge Version control directory and linked into the TeamForge Service Request

<sup>2</sup> Sonar tool will be used for software quality verification

- Assess and assure **Source Code Maintainability** (Understandability, Modifiability, Traceability, Testability, Portability, and Reusability) to identify issues/potential problems, propose solutions/changes and confirm implementation results.
- Gathering and interpret Source Code metrics.

In order to carry out this quality verification the following must be done:

- Perform Earlier Verifications:
  - focus on evaluating intermediate software builds and removing defects at coding time. SONAR is the tool selected for this activity. The Service Transition Team will inform in due time the Contractors on the criteria for testing and about the use of SONAR to perform this verification.
- Execute Test Plan:
  - focuses on executing tests cases/procedures approved.
- Perform Post-Product Verifications:
  - focus on evaluating final build quality or finding defect root causes after the product is complete.

Table 1 shows what tests have to be carried out in different environments although the MAT (Service Transition Team) is responsible for defining the profile of the testing activities considering the nature of the changes in the application and/or infrastructure.

**Table 1. Environments vs. Test Types**

TYPE OF TESTS	ENVIRONMENTS		
	TEST	PRE-PRODUCTION	PRODUCTION
SMOKE	Yes	Yes	Yes
FUNCTIONAL	Yes	Yes	
NON FUNCTIONAL		Yes	
ACCEPTANCE		Yes	

When applicable, the release will be deployed first in the Test environment in order to be functional tested.

If the results are successful then the release will be deployed in the Pre-Production environment to complete a second round of functional and technical testing. All deployments will be executed as described in the release documentation. Annex 5 shows an example of traceability matrix.

The test results are collected, analysed, and compared to the expected results in order to draft the Test Report (Annex 4, Template III). All test iterations and results will be included in Test Report and filled-in during the tests execution.

The complete text of the SVVT could be provided to the successful tenderer on request via e-mail following the kick-off meeting of the contract.

## SVVT procedure, ANNEX 4

**TEMPLATES****I. Test Plan (Standard IEEE 829-1998)****0. Unique Identification of the Test Plan****1. Introduction**

- 1.1 Summary of Features and Objects to be tested
- 1.2 References
- 1.3 Definitions and Acronyms

**2. Tests Objects**

- 2.1 Objects and respective test versions
- 2.2 Related documents
- 2.3. Bug Reports related to the objects to be tested
- 2.4 Objects no to be tested

**3. Features to be tested (functional and non-functional)**

- 3.1. All features and combinations to be tested
- 3.2 Related documents (Use and Test Cases with expected results)

**4 Features not to be tested (functional and non-functional)**

- 4.1 All features and combinations to not be tested
- 4.2 Rationale

**5. Approach**

- 5.1 General Testing Approach (type of tests)
- 5.2 Testing Approach for group of features/combination of features
- 5.3 Activities, techniques, and tools to be used

**6. Limitations****7. Not successful Test Criteria****8. Suspension Test Criteria and requirements for re-starting**

- 8.1 Specification of criteria to suspend tests
- 8.2 Specification of activities to repeat tests

**9. Test Tasks**

- 9.1 Identification of tasks / procedures to prepare and execute tests
- 9.2 Identification of interconnections tasks

**10. Test Environment**

- 10.1 Specifications of the test environment
- 10.2 Specifications of the level of security
- 10.3. Specifications of special testing tools
- 10.4 Other needs required

**11. Roles and Responsibilities** (developers, testers, business team, A.3)**12. Knowledge and Training Requirements****13. Schedule****14. Risk Management****15. Approval**

## **II. Test Cases Specification (Standard IEEE 829-1998)**

### **0. Unique Identification of the Test Case**

#### **1. Test Objects**

- 1.1 Specification and conditions of the features to be tested

#### **2 Input Specifications**

- 2.1 Data Identification
- 2.2 Ordering
- 2.3. Values
- 2.3 States
- 2.4 Timing

#### **3 Output Specifications**

- 3.1 Data Identification
- 3.2 Ordering
- 3.3. Values
- 3.3 States
- 3.4 Timing

### **4. Special Procedures**

### **5. Dependencies**

## **III. Test Report**

### **1. Executive Summary**

- 1.1. Guidelines
- 1.2. Glossary

### **2. Testing Tools**

### **3. Objectives**

### **4. Test Environment**

- 4.1. Description
- 4.2. Diagrams
- 4.3. Servers

### **5. Scripts**

### **6. Test Execution**

- 6.1. Execution Process
- 6.2. Sequence of Scenarios
- 6.3. Tasks between executions

### **7. Result Analysis**

### **8. Conclusions and Recommendations**