INTERNATIONAL STANDARD

ISO/IEC/ IEEE 29119-3

First edition
2013-09-01

Software and systems engineering — Software testing —

Part 3: Test documentation

Ingénierie du logiciel et des systèmes — Essais du logiciel —

Partie 3: Documentation des essais

Reference number
ISO/IEC/IEEE 29119-3:2013(E)

ISO copyright office
Case postale 56
CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

IEC Central Office
3, rue de Varembé
CH-1211 Geneva 20
Switzerland
E-mail inmail@iec.ch
Web www.iec.ch

Institute of Electrical and Electronics Engineers, Inc.
3 Park Avenue, New York
NY 10016-5997, USA
E-mail stds.ipr@ieee.org
Web www.ieee.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

IEEE Standards documents are developed within the IEEE Societies and the Standards Coordinating Committees of the IEEE Standards Association (IEEE-SA) Standards Board. The IEEE develops its standards through a consensus development process, approved by the American National Standards Institute, which brings together volunteers representing varied viewpoints and interests to achieve the final product. Volunteers are not necessarily members of the Institute and serve without compensation. While the IEEE administers the process and establishes rules to promote fairness in the consensus development process, the IEEE does not independently evaluate, test, or verify the accuracy of any of the information contained in its standards.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of ISO/IEC JTC 1 is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is called to the possibility that implementation of this standard may require the use of subject matter covered by patent rights. By publication of this standard, no position is taken with respect to the existence or validity of any patent rights in connection therewith. ISO/IEEE is not responsible for identifying essential patents or patent claims for which a license may be required, for conducting inquiries into the legal validity or scope of patents or patent claims or determining whether any licensing terms or conditions provided in connection with submission of a Letter of Assurance or a Patent Statement and Licensing Declaration Form, if any, or in any licensing agreements are reasonable or non-discriminatory. Users of this standard are expressly advised that determination of the validity of any patent rights, and the risk of infringement of such rights, is entirely their own responsibility. Further information may be obtained from ISO or the IEEE Standards Association.

ISO/IEC/IEEE 29119-3 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 7, *Software and systems engineering*, in cooperation with the Software & Systems Engineering Standards Committee of the IEEE Computer Society, under the Partner Standards Development Organization cooperation agreement between ISO and IEEE.

ISO/IEC 29119 consists of the following standards, under the general title *Software and systems engineering — Software testing*:

— *Part 1: Concepts and definitions*

— *Part 2: Test processes*

— *Part 3: Test documentation*

— *Part 4: Test techniques*

Introduction

The purpose of the ISO/IEC/IEEE 29119 series of software testing standards is to define an internationally-agreed set of standards for software testing that can be used by any organization when performing any form of software testing.

This part of ISO/IEC/IEEE 29119, Test Documentation, includes templates and examples of test documentation that are produced during the test process. The templates are arranged within clauses reflecting the overall test process description structure in ISO/IEC/IEEE 29119-2 Test Processes, i.e. by the test process in which they are being produced. Annex A contains outlines of the contents of each document. Annex B contains a list of all the information items identified in Clauses 5, 6 and 7 of this part of ISO/IEC/IEEE 29119 with the corresponding level of conformance (shall/should/may) from ISO/IEC/IEEE 29119-2 Test Processes. Annex C contains an overview of the examples. Annexes D to S contain examples of the application of the templates. Annex T provides mappings to existing standards. The Bibliography for this part of ISO/IEC/IEEE 29119 is at the end of the document.

The concepts and vocabulary relating to the software testing documentation are defined in ISO/IEC/IEEE 29119-1 Concepts and Definitions.

The actual test process model is defined in ISO/IEC/IEEE 29119-2 Test Processes. It comprises test process descriptions that define the software testing processes at the organizational level, test management level and dynamic test level. Supporting informative diagrams describing the processes are also provided.

Software test design techniques that can be used during test design are defined in ISO/IEC/IEEE 29119-4 Test Techniques.

This series of international standards aims to provide stakeholders with the ability to manage and perform software testing in any organization.

Software and systems engineering — Software testing —

Part 3: Test documentation

1 Scope

This part of ISO/IEC/IEEE 29119 specifies software test documentation templates that can be used by any organization, project or smaller testing activity. It describes the test documentation that is an output of the processes specified in ISO/IEC/IEEE 29119-2 Test Processes. An overview of the documents is provided in Figure 1 below. A slightly larger version of this figure is provided in Annex A.

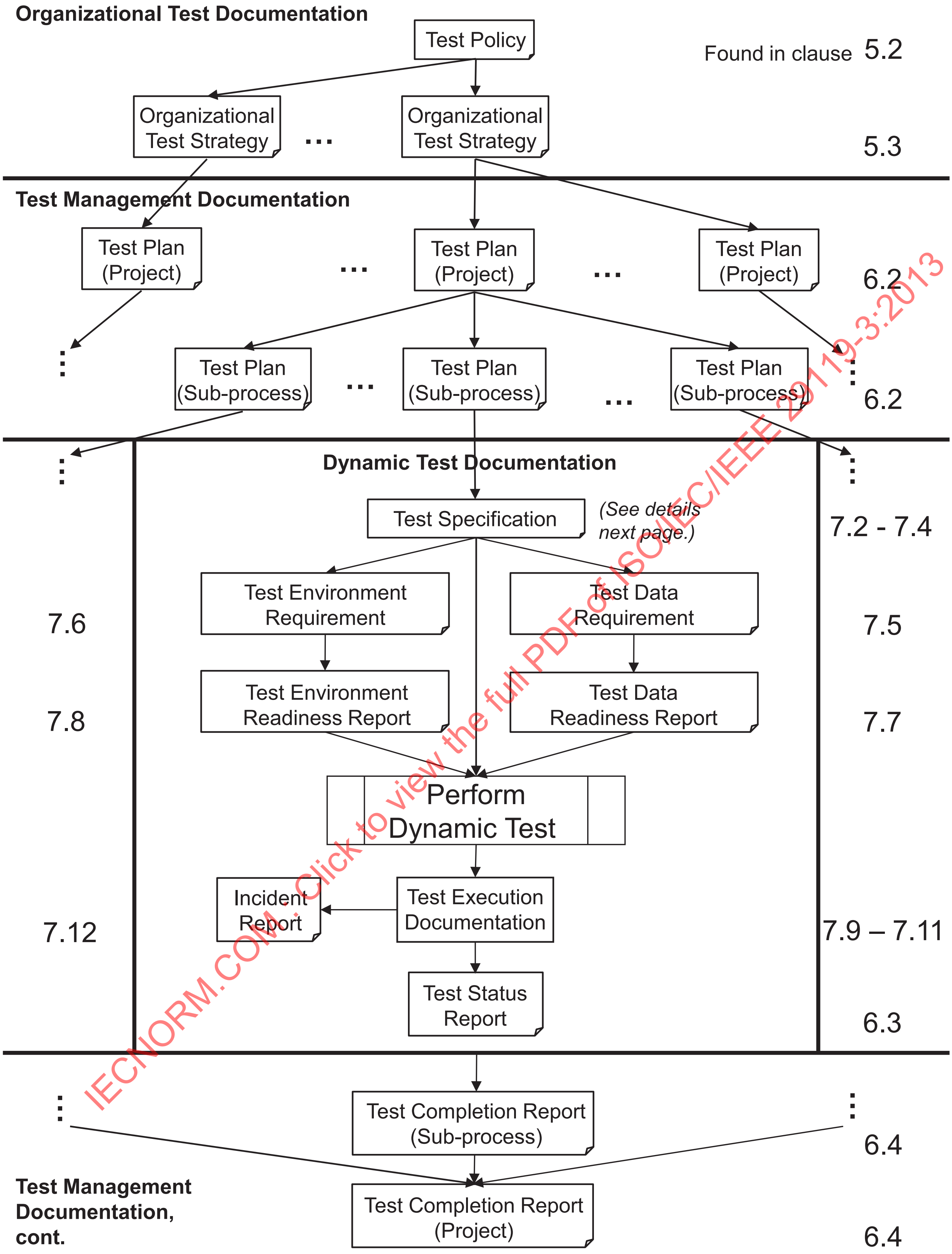

Organizational Test Documentation
Test Policy
Found in clause 5.2
Organizational Test Strategy
...
Organizational Test Strategy
5.3
Test Management Documentation
Test Plan (Project)
...
Test Plan (Project)
...
Test Plan (Project)
6.2
Test Plan (Sub-process)
...
Test Plan (Sub-process)
...
Test Plan (Sub-process)
6.2
Dynamic Test Documentation
Test Specification
(See details next page.)
7.2 - 7.4
Test Environment Requirement
7.6
Test Data Requirement
7.5
Test Environment Readiness Report
7.8
Test Data Readiness Report
7.7
Perform Dynamic Test
Incident Report
7.12
Test Execution Documentation
7.9 – 7.11
Test Status Report
6.3
Test Completion Report (Sub-process)
6.4
Test Management Documentation, cont.
Test Completion Report (Project)
6.4

Figure 1 — The hierarchy of test documentation

This part of ISO/IEC/IEEE 29119 is applicable to testing in all software development lifecycle models.

This part of ISO/IEC/IEEE 29119 is intended for, but not limited to, testers, test managers, developers, and project managers, particularly those responsible for governing, managing, and implementing software testing.

The documents described in this part of ISO/IEC/IEEE 29119 may be issued in several versions over time. However, the handling of multiple versions of documents is out of scope of this part of ISO/IEC/IEEE 29119, because this is a configuration management issue.

2 Conformance

2.1 Intended usage

The requirements in this part of ISO/IEC/IEEE 29119 are contained in Clauses 5, 6 and 7. This part of ISO/IEC/IEEE 29119 provides requirements for a number of test documents suitable for use during the complete software lifecycle. It is recognized that particular projects or organizations may not need to use all of the documents defined by this part of ISO/IEC/IEEE 29119. Therefore, implementation of this part of ISO/IEC/IEEE 29119 typically involves selecting a set of documents suitable for the organization or project. There are two ways that an organization can claim to conform to the provisions of this part of ISO/IEC/IEEE 29119; full or tailored conformance. Conformance may be claimed for organizations, projects, multi-supplier projects and services, as identified in the claim of conformance.

The information items identified in Clauses 5, 6, and 7 of this part of ISO/IEC/IEEE 29119 correspond to the outputs of the ISO/IEC/IEEE 29119-2 Test Processes. Annex B is normative and provides an overview of the normative requirements for the clauses in ISO/IEC/IEEE 29119-2 where the creation of the information items defined in Clauses 5, 6, and 7 of this part of ISO/IEC/IEEE 29119 is described.

In this part of ISO/IEC/IEEE 29119, for simplicity of reference, each document is described as if it were published as a separate hardcopy document. Document titles and contents provided in this part of ISO/IEC/IEEE 29119 may be modified (added to, combined or re-titled) and use of the nomenclature of the specific records in Clauses 5, 6 and 7 is not required to claim conformance. Documents shall be considered as conforming if they are unpublished but available in electronic form, divided into separate documents or volumes, or combined with other documents into one document.

2.2 Types of conformance

The following types of conformance shall be asserted. The selected type shall be identified in the claim of conformance documentation.

2.2.1 Full Conformance

The minimum set of required information items is all of those information items specified in Clauses 5, 6 and 7 of this part of ISO/IEC/IEEE 29119.

NOTE Full conformance could be claimed for selected documents even if full conformance with the entire standard is not claimed.

2.2.2 Tailored Conformance

The content of the test documents defined in Clauses 5, 6 and 7 of this part of ISO/IEC/IEEE 29119 may be tailored based on the tailored conformance to ISO/IEC/IEEE 29119-2 Test Processes and/or based on the specific needs of an organization or project. Where tailoring occurs, justification shall be provided whenever an information item defined in Clauses 5, 6 and 7 of this part of ISO/IEC/IEEE 29119 is not prepared. All tailoring decisions shall be recorded with their rationale, including the consideration of any applicable risks. Tailoring decisions shall be agreed by the relevant stakeholders.

Tailored conformance can be achieved by:

a) The minimum set of required test documentation is determined by the tailoring of the processes and activities in accordance with Clause 2 of ISO/IEC/IEEE 29119-2 Test Processes; and/or

b) The minimum set of required test documentation is determined according to specific organization and/or project needs; and/or

c) The minimum set of required information items within documents is determined according to specific organization and/or project needs.

NOTE 1 In projects, particularly those following an agile approach, tailoring can be applied to all Part 3 documents to allow them to be condensed or presented in an alternate format (e.g. verbal or slide presentation).

NOTE 2 Different document names could be used, but when this is done and conformity needs to be demonstrated, a mapping is often produced between this part of ISO/IEC/IEEE 29119 and local usage to aid conformity assessment.

3 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC/IEEE 15289:2011, *Systems and software engineering — Content of life-cycle information products (documentation)*

ISO/IEC/IEEE 29119-1, *Software and systems engineering — Software testing — Part 1: Concepts and definitions*

ISO/IEC/IEEE 29119-2, *Software and systems engineering — Software testing — Part 2: Test processes*

Other standards useful for the implementation and interpretation of this standard are listed in the Bibliography.

4 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC/IEEE 24765 and the following apply.

NOTE Use of the terminology in this part of ISO/IEC/IEEE 29119 is for ease of reference and is not mandatory for conformance with this part of ISO/IEC/IEEE 29119. The following terms and definitions are provided to assist with the understanding and readability of this part of ISO/IEC/IEEE 29119. Only terms critical to the understanding of this part of ISO/IEC/IEEE 29119 are included. This clause is not intended to provide a complete list of testing terms. The Systems and Software Engineering vocabulary ISO/IEC/IEEE 24765 can be referenced for terms not defined in this clause. All terms defined in this clause are also intentionally included in ISO/IEC/IEEE 29119-1, as that international standard includes all terms that are used in ISO/IEC/IEEE 29119-1, 2, 3 and 4.

4.1
actual results
set of behaviours or conditions of a test item, or set of conditions of associated data or the test environment, observed as a result of test execution

EXAMPLE Output to screen, outputs to hardware, changes to data, reports, and communication messages sent.

4.2
coverage item
see test coverage item (4.15)

4.3
expected results
observable predicted behaviour of the test item under specified conditions based on its specification or another source

4.4
feature set
logical subset of the test item(s) that could be treated independently of other feature sets in the subsequent test design activities

Note 1 to entry: This could be the set of all features for the item (its full feature set), or a subset identified for a specific purpose (the functional feature set, etc.).

4.5
Incident Report
documentation of the occurrence, nature, and status of an incident

Note 1 to entry: Incident reports are also known as anomaly reports, bug reports, defect reports, error reports, issues, problem reports and trouble reports, amongst other terms.

4.6
Organizational Test Specification
document that provides information about testing for an organization, i.e. information that is not project specific

EXAMPLE The most common examples of organizational test specifications are the Organizational Test Policy and Organizational Test Strategy.

4.7
Organizational Test Strategy
document that expresses the generic requirements for the testing to be performed on all the projects run within an organization, providing detail on how the testing is to be performed

Note 1 to entry: The Organizational Test Strategy is aligned with the Organizational Test Policy.

Note 2 to entry: An organization could have more than one Organizational Test Strategy to cover markedly different project contexts.

Note 3 to entry: The Organizational Test Strategy could incorporate the context of the Test Policy where no separate Test Policy is available

4.8
product risk
risk that a product could be defective in some specific aspect of its function, quality, or structure

4.9
project risk
risk related to the management of a project

EXAMPLE Lack of staffing, strict deadlines, changing requirements.

4.10
regression testing
testing following modifications to a test item or to its operational environment, to identify whether regression failures occur

Note 1 to entry: The sufficiency of a set of regression test cases depends on the item under test and on the modifications to that item or its operational environment.

4.11
retesting
re-execution of test cases that previously returned a "fail" result, to evaluate the effectiveness of intervening corrective actions

Note 1 to entry: Also known as confirmation testing.

4.12
test case
set of test case preconditions, inputs (including actions, where applicable), and expected results, developed to drive the execution of a test item to meet test objectives, including correct implementation, error identification, checking quality, and other valued information

Note 1 to entry: A test case is the lowest level of test input (i.e. test cases are not made up of test cases) for the test sub-process for which it is intended.

Note 2 to entry: Test case preconditions include: test environment, existing data (e.g. databases), software under test, hardware, etc.

Note 3 to entry: Inputs are the data information used to drive test execution.

Note 4 to entry: Expected results include success criteria, failures to check for, etc.

4.13
Test Case Specification
documentation of a set of one or more test cases

4.14
Test Completion Report
report that provides a summary of the testing that was performed

Note 1 to entry: Also known as a Test Summary Report.

4.15
test coverage item
attribute or combination of attributes that is derived from one or more test conditions by using a test design technique that enables the measurement of the thoroughness of the test execution

4.16
test data
data created or selected to satisfy the input requirements for executing one or more test cases, which could be defined in the Test Plan, test case or test procedure

Note 1 to entry: Test data could be stored within the product under test (e.g., in arrays, flat files, or a database), or could be available from or supplied by external sources, such as other systems, other system components, hardware devices, or human operators.

4.17
Test Data Readiness Report
document describing the status of each test data requirement

4.18
Test Design Specification
document specifying the features to be tested and their corresponding test conditions

4.19
test design technique
activities, concepts, processes, and patterns used to construct a test model that is used to identify test conditions for a test item, derive corresponding test coverage items, and subsequently derive or select test cases

4.20
test environment
facilities, hardware, software, firmware, procedures, and documentation intended for or used to perform testing of software

Note 1 to entry: A test environment could contain multiple environments to accommodate specific test sub-processes (e.g. a unit test environment, a performance test environment, etc.).

4.21
test environment readiness report
document that describes the fulfillment of each test environment requirement

4.22
Test Environment Requirements
description of the necessary properties of the test environment

Note 1 to entry: All or parts of the test environment requirements could reference where the information can be found, e.g. in the appropriate Organizational Test Strategy, Test Plan, and/or Test Specification.

4.23
Test Execution Log
document that records details of the execution of one or more test procedures

4.24
test item
work product that is an object of testing

EXAMPLE A system, a software item, a requirements document, a design specification, a user guide.

4.25
Test Plan
detailed description of test objectives to be achieved and the means and schedule for achieving them, organized to coordinate testing activities for some test item or set of test items

Note 1 to entry: A project could have more than one Test Plan, for example there could be a Project Test Plan (also known as a Master Test Plan) that encompasses all testing activities on the project; further detail of particular test activities could be defined in one or more test sub-process plans (e.g. a system test plan or a performance test plan).

Note 2 to entry: Typically a Test Plan is a written document, though other formats could be possible as defined locally within an organization or project.

Note 3 to entry: Test Plans could also be written for non-project activities, for example a Maintenance Test Plan.

4.26
Test Policy
an executive-level document that describes the purpose, goals, principles and scope of testing within an organization

Note 1 to entry: The Test Policy defines what testing is performed and what it is expected to achieve but does not detail how testing is to be performed.

Note 2 to entry: The Test Policy can provide a framework for establishing, reviewing and continually improving the organisations testing.

4.27
Test Procedure Specification
document specifying one or more test procedures, which are collections of test cases to be executed for a particular objective

Note 1 to entry: The test cases in a test set are listed in their required order in the test procedure.

Note 2 to entry: Also known as a manual test script. A test procedure specification for an automated test run is usually called a test script.

4.28
test result
indication of whether or not a specific test case has passed or failed, i.e. if the actual results correspond to the expected results or if deviations were observed

4.29
test set
collection of test cases for the purpose of testing a specific test objective

Note 1 to entry: The test sets will typically reflect the feature sets, but they could contain test cases for a number of feature sets.

Note 2 to entry: Test cases for a test set could be selected based on the identified risks, test basis, retesting and/or regression testing.

4.30
test specification
complete documentation of the test design, test cases and test procedures for a specific test item

Note 1 to entry: A Test Specification could be detailed in one document, in a set of documents, or in other ways, for example in a mixture of documents and database entries.

4.31
test status report
report that provides information about the status of the testing that is being performed in a specified reporting period

4.32
test strategy
part of the Test Plan that describes the approach to testing for a specific test project or test sub-process or sub-processes

Note 1 to entry: The test strategy is a distinct entity from the Organizational Test Strategy.

Note 2 to entry: The test strategy usually describes some or all of the following: the test practices used; the test sub-processes to be implemented; the retesting and regression testing to be employed; the test design techniques and corresponding test completion criteria to be used; test data; test environment and testing tool requirements; and expectations for test deliverables.

4.33
test traceability matrix
document, spreadsheet, or other automated tool used to identify related items in documentation and software, such as requirements with associated tests

Note 1 to entry: Also known as: verification cross reference matrix, requirements test matrix, requirements verification table, and others.

Note 2 to entry: Different test traceability matrices could have different information, formats, and levels of detail.

4.34
testing
set of activities conducted to facilitate discovery and/or evaluation of properties of one or more test items

Note 1 to entry: Testing activities include planning, preparation, execution, reporting, and management activities, insofar as they are directed towards testing.

5 Organizational Test Process Documentation

5.1 Overview

Organizational test specifications describe information about testing at the organization level, and are not project-dependent. Typical examples of organizational test specifications developed in the organizational test process include:

- Test Policy;
- Organizational Test Strategy.

The full templates with explanatory text for the document follow in sub-clauses 5.2 Test Policy and 5.3 Organizational Test Strategy. Annex A provides an abbreviated overview of each document. Annexes D and E provide examples of a Test Policy and Organizational Test Strategy for example projects.

5.2 Test Policy

5.2.1 Overview

The Test Policy defines the objectives and principles of software testing to be applied within the organization. It defines what should be accomplished by testing, but does not detail how testing is performed. The policy provides a framework for establishing, reviewing, and continually improving the organization’s Test Policy.

Annex A.2.2 provides an outline of the Organizational Test Policy, while annexes D.1 and D.2 provide examples that demonstrate how Organizational Test Policies could be developed for two different example projects.

The contents of the Test Policy include:

5.2.2 Document specific information

5.2.2.1 Overview

This information identifies the document and describes its origins and history.

NOTE The information could be placed on an early page in a document, or in a central place, if the contents are kept in electronic form, e.g. in a database.

5.2.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

5.2.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

5.2.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

5.2.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change could be document author, project manager, system owner.

5.2.3 Introduction

Provides explanatory information about the context and structure of the document.

5.2.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

5.2.3.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into "external" references that are imposed from outside the organization and "internal" references that are imposed from within the organization.

EXAMPLE Referenced documents could be policies, plans, procedures, and other source data.

5.2.3.3 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

5.2.4 Test policy statements

5.2.4.1 Objectives of testing

Describes the purpose, goals, and overall scope of the testing within the organization. States the organization's position on why testing is performed and what they look to achieve.

5.2.4.2 Test process

Identifies the test process that the organization will follow. This may include reference to a specific document providing details of the test process.

EXAMPLE Such a document could be ISO/IEC/IEEE 29119-2 Test Processes. The details of activities in the test process could be described in more detailed test process documentation.

5.2.4.3 Test organization structure

Identifies the roles and structure of the test organization. A diagram to show test organization hierarchy may be used, or the information may be presented in a table.

5.2.4.4 Tester training

States required training and certifications for individuals working in the test organization.

5.2.4.5 Tester ethics

Identifies the organizational ethics code to be upheld by the testers.

5.2.4.6 Standards

States which standards are applicable within the test organization.

5.2.4.7 Other relevant policies

Identifies policies that impact the test organization.

EXAMPLE A policy statement could be that testing will conform to the Quality Policy.

5.2.4.8 Measuring the value of testing

States how the organization determines the return on investment of testing. Identifies the objectives for measuring the value of testing.

5.2.4.9 Test asset archiving and reuse

States the organization's position on the archiving and reuse of test assets.

5.2.4.10 Test process improvement

States the method for ensuring continuous improvement of the test process.

5.3 Organizational Test Strategy

5.3.1 Overview

The Organizational Test Strategy is a technical document that provides guidelines on how testing should be carried out within the organization, i.e. how to achieve the objectives stated in the Test Policy.

The Organizational Test Strategy is a generic document at an organizational level that provides guidelines to projects within its scope; it is not project-specific.

For small or highly homogenous organizations a single Organizational Test Strategy may cover all testing activities. An organization may have more than one Organizational Test Strategy if the organization performs development in a number of significantly different ways, such as both safety-critical products and non-critical products; or if it is using both agile and V-model development models; or if its programmes are large enough to merit their own strategy.

The Organizational Test Strategy may incorporate the content of the Test Policy where no separate Test Policy is available.

An Organizational Test Strategy includes identification of relevant test sub-processes and strategy statements for each of these. The document may be partitioned with a sub-section for each of the identified test sub-process if the test sub-processes strategy statements differ significantly between the test sub-processes; this is illustrated in the figure below.

Annex A.2.3 provides an outline of the Organizational Test Strategy, while annexes E.1 and E.2 provide examples that demonstrate how Organizational Test Strategies could be developed for two different example projects.

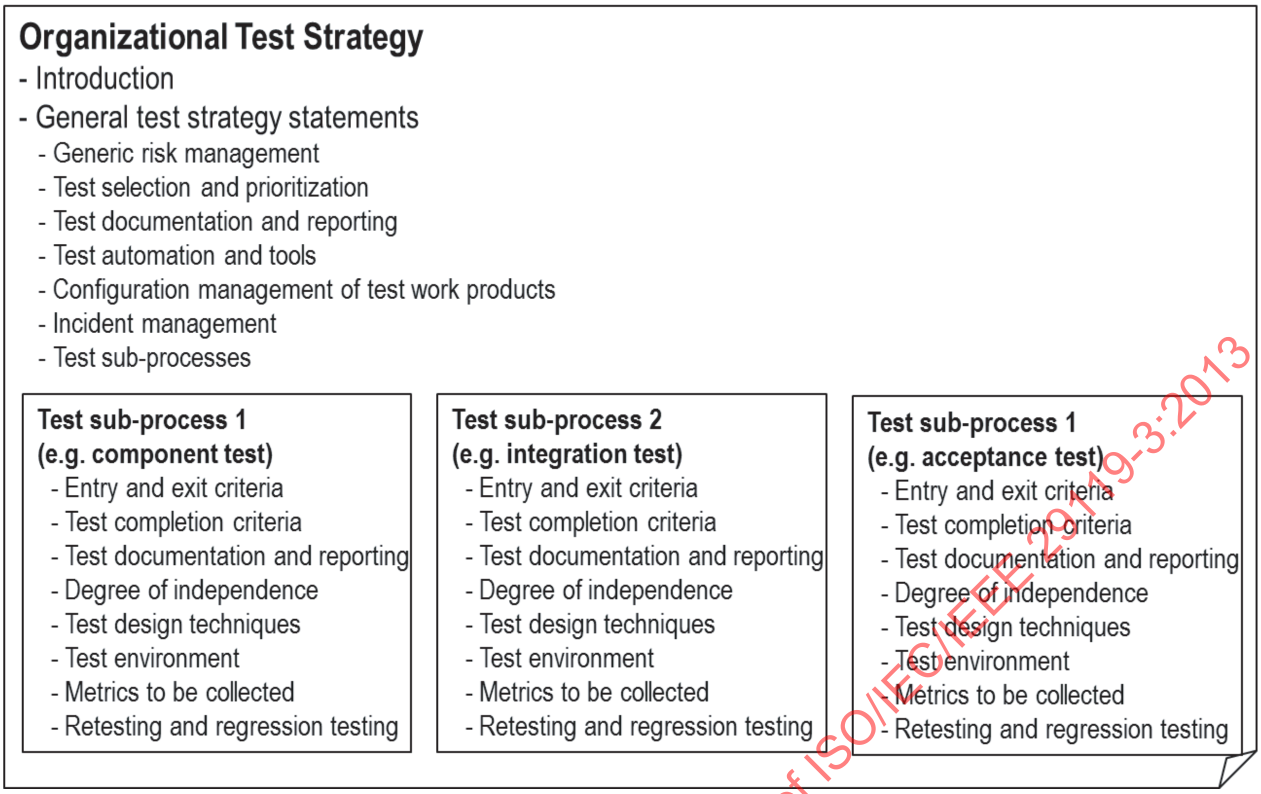

Organizational Test Strategy
- Introduction
- General test strategy statements
- Generic risk management
- Test selection and prioritization
- Test documentation and reporting
- Test automation and tools
- Configuration management of test work products
- Incident management
- Test sub-processes
Test sub-process 1
(e.g. component test)
- Entry and exit criteria
- Test completion criteria
- Test documentation and reporting
- Degree of independence
- Test design techniques
- Test environment
- Metrics to be collected
- Retesting and regression testing
Test sub-process 2
(e.g. integration test)
- Entry and exit criteria
- Test completion criteria
- Test documentation and reporting
- Degree of independence
- Test design techniques
- Test environment
- Metrics to be collected
- Retesting and regression testing
Test sub-process 1
(e.g. acceptance test)
- Entry and exit criteria
- Test completion criteria
- Test documentation and reporting
- Degree of independence
- Test design techniques
- Test environment
- Metrics to be collected
- Retesting and regression testing

Figure 2 — An example structure of the Organizational Test Strategy

The contents of the Organizational Test Strategy include:

5.3.2 Document specific information

5.3.2.1 Overview

This information identifies the document and describes its origins and history.

NOTE The information could be placed on an early page in a document, or in a central place, if the contents are kept in electronic form, e.g. in a database.

5.3.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

5.3.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

5.3.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

5.3.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change could be document author, project manager, system owner.

5.3.3 Introduction

Provides explanatory information about the context and structure of the document.

5.3.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

5.3.3.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into “external” references that are imposed from outside the organization and “internal” references that are imposed from within the organization.

EXAMPLE Documents could be policies, plans, procedures, and other source data.

5.3.3.3 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

5.3.4 Project-wide organizational test strategy statements

The strategy is defined for the specified scope. This section includes statements that are appropriate for all test sub-processes to be performed in a given project within the scope of the strategy. This section may include subsections from the policy, if needed.

5.3.4.1 Generic risk management

Identifies the generic approach to risk management expected to be used to direct the testing activities.

5.3.4.2 Test selection and prioritization

Describes the organization’s approach to selecting and prioritizing test execution, in the form of prioritized test procedures. Test procedures consist of prioritized test cases, derived from prioritized feature sets via prioritized test conditions and coverage items.

5.3.4.3 Test documentation and reporting

Identifies the documents expected to be produced during testing for the test project as a whole. Describes when each is prepared and the associated approval process. This is tightly connected to the test process that is specified in the policy.

5.3.4.4 Test automation and tools

Describes the approach to test automation within the organization. Identifies the testing tools to be used throughout testing.

EXAMPLE This could include test management tools, test execution tools, performance testing tools, security testing tools, usability testing tools.

5.3.4.5 Configuration management of test work products

Describes the configuration management to be performed for the work products from testing; describes how these work products are to be identified, traced, stored, and made available to stakeholders.

5.3.4.6 Incident management

Describes how incidents should be managed during testing or refers to a description elsewhere.

5.3.4.7 Test sub-processes

Identifies specific test sub-processes to be performed as part of the testing within the scope of the strategy.

5.3.5 Test sub-process-specific organizational test strategy statements

5.3.5.1 Entry and exit criteria

Specifies the criteria used to determine when the test activities for the defined test sub-process should start and stop.

A test sub-process consists of the following processes:

- Test design & implementation;
- Test environment set-up & maintenance;
- Test execution; and
- Test incident reporting.

Different entry and exit criteria may be defined for each of these individually, or for selected ones, or for the entire sub-process as a whole.

5.3.5.2 Test completion criteria

Describes how the organization considers the testing activities for the test sub-process to be complete.

5.3.5.3 Test documentation and reporting

Identifies the test documentation, including reporting, used for testing activities in the test sub-process. Describes when each document or report is prepared and the associated approval process. This is tightly connected to the test process specified in the policy.

5.3.5.4 Degree of independence

Establishes the level of independence of those performing the testing. States how this testing group is technically, managerially, and financially independent.

5.3.5.5 Test design techniques

Identifies specific test design techniques to be used during test design and implementation within the test sub-process.

5.3.5.6 Test environment

Identifies the test environment for the test sub-process; may state where specific types of test should be performed, and identifies groups or organizations responsible for the test environment. May identify the origin of test data, state where particular types of test data are located, and groups or organizations responsible for test data.

5.3.5.7 Metrics to be collected

Describes the metrics for which values are to be collected during the test activities in the test sub-process.

5.3.5.8 Retesting and regression testing

Identifies the strategy, conditions, and activities for retesting and regression testing in the test sub-process.

6 Test Management Processes Documentation

6.1 Overview

The documents developed in the test management processes comprise the following types:

- Test Plan;
- Test Status Report;
- Test Completion Report.

The full templates with explanatory text for the documents are found below. Annex A provides an abbreviated overview of each document. Annexes F, G, and H provide examples of Test Plans, Test Status Reports, and Test Completion Reports for example projects.

6.2 Test Plan

6.2.1 Overview

The Test Plan provides a test planning and test management document. Some projects may have a single test plan, while for larger projects multiple test plans may be produced. Test plans may apply across multiple projects (at the programme level), or to a single project (project test plan/master test plan), or to a specific test sub-process (system test plan, integration software test plan, sub-system test plan, sub-contractor software test plan, unit software test plan, or performance test plan, or to a specific iteration of testing). If more software test plans are created, a mapping tree may be produced to aid documenting relationships and the information contained in each.

The Test Plan describes the decisions made during the initial planning and evolves as re-planning is performed as part of the control activity.

Annex A.2.4 provides an outline of the Test Plan, while annexes F.1 and F.2 provide examples that demonstrate how Test Plans could be developed for two different example projects.

The contents of the Test Plan include:

6.2.2 Document specific information

6.2.2.1 Overview

This information identifies the document and describes its origins and history.

NOTE The information could be placed on an early page in a document, or in a central place if the contents are kept in electronic form, e.g. in a database.

6.2.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

6.2.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

6.2.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

6.2.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change could be document author, project manager, system owner.

6.2.3 Introduction

Provides explanatory information about the context and structure of the document.

6.2.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

6.2.3.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into “external” references that are imposed from outside the organization and “internal” references that are imposed from within the organization.

EXAMPLE References to test item documentation relevant to an individual test sub-process could include:

- Requirements;
- Design;
- User’s guide;
- Operations guide; and/or
- Installation guide.

6.2.3.3 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

6.2.4 Context of the testing

6.2.4.1 Project(s) / test sub-process(es)

Identifies the project(s) or the test sub-process(es) for which the plan is being written and other relevant contextual information.

6.2.4.2 Test item(s)

Identifies the test item(s) for the testing covered by this plan including their version/revision or reference where this information can be found.

This section may describe the mission/business purpose of the test item(s), or reference where this information can be found.

NOTE This information could be defined in a system definition document, such as a concepts of operations.

EXAMPLE The test item could be a software unit, interfaces between units, a subsystem, or a complete system.

It may also identify any procedures for the transfer of the test item(s) from other environments to the test environment.

6.2.4.3 Test scope

Summarizes the features of the test item(s) to be tested. Also identifies any features of the test item(s) that are to be specifically excluded from testing and the rationale for their exclusion.

EXAMPLE Features to be tested could be specific attributes of the software, functions, interfaces, or business processes.

6.2.4.4 Assumptions and constraints

Describes any assumptions and constraints for the test effort covered by this plan. These may include regulatory standards, the requirements in the Test Policy and the Organizational Test Strategy, contractual requirements, project time and cost constraints, and availability of appropriately-skilled staff, tools and/or environments.

6.2.4.5 Stakeholders

Lists the stakeholders and their relevance to the testing. Describes how the communication with each stakeholder is to be performed.

6.2.5 Testing communication

Describes the lines of communication between testing, other lifecycle activities, and within the organization.

EXAMPLE This could include the authority for resolving issues raised as a result of the testing activities and the authority for approving test products and processes.

This information may be represented visually.

NOTE A visual representation could include an organization chart or a figure that illustrates the flow of information and data.

6.2.6 Risk register

Identifies the risks considered by the testing covered by this plan. This should include any relevant risks that may be specified in the Organizational Test Strategy. Provides an exposure level for each risk based on its impact and probability. Provides recommendations to treat the risks. This section may reference where a separate risk register can be found.

EXAMPLE Recommendations to treat risk could include eliminate, reduce, or ignore risk.

NOTE A risk register could be located in a project plan or a risk management plan.

6.2.6.1 Product risks

Identifies test-related product risks and provides recommendations to treat each risk.

EXAMPLE Test-related product risks could include defects in functionality or in non-functional aspects such as performance.

6.2.6.2 Project risks

Identifies test-related project risks and provides recommendations to treat each risk.

EXAMPLE Test-related project risks could include risks related to schedule or resources.

6.2.7 Test strategy

Describes the approach to testing for the specified test project or test sub-process, as outlined in the following sub-clauses. The document may refer to the Organization Test Strategy stating only its differences from it.

6.2.7.1 Test sub-processes

For a project test plan this identifies the sub-processes of testing that will be conducted.

6.2.7.2 Test deliverables

Identifies all documents that are to be delivered from the testing activity or equivalent information to be recorded electronically, for example in databases or dedicated test tools.

EXAMPLE The following documents could be included:

- Test Plan;
- Test Design Specification;
- Test Case Specification;
- Test Procedure Specification;
- Test Data Readiness Report;
- Test Environment Readiness Report;
- Incident Reports;
- Test Status Reports; and
- Test Completion Report.

Test input data and test output data may be identified as deliverables. Test tools created as part of the testing activity may also be included. If documents have been combined or eliminated, then this list will be modified accordingly.

This subsection may include when the document(s) should be delivered, and to/from whom (preferably by position, not name).

6.2.7.3 Test design techniques

Specifies which test design techniques are to be applied.

6.2.7.4 Test completion criteria

Describes the conditions under which the relevant test organization considers test execution activities to be complete.

EXAMPLE This could be when a specific coverage has been reached and the number of outstanding defects is under a specified limit.

6.2.7.5 Metrics to be collected

Describes the metrics for which values are to be collected during the test activities.

6.2.7.6 Test data requirements

Specifies all relevant test data requirements for the project or test sub-process (as appropriate).

EXAMPLE This could identify the origin of the test data and state where specific test data is located, whether data has to be disguised for confidentiality reasons, and/or the role responsible for the test data.

These test data requirements may be deferred to the Test Data Requirements document (see 7.5), as applicable.

6.2.7.7 Test environment requirements

Specifies the necessary and desired properties of the test environment.

EXAMPLE This could include hardware, software, testing tools, databases, and personnel (identifying their organizations, as appropriate).

Includes information regarding selection, evaluation, acquisition and support for each tool. It may include test environment requirements for test preparation, test execution (including data capture), and any post-execution activities.

EXAMPLE A post-execution activity could be data analysis.

These test environment requirements may be deferred to the Test Environment Requirements document (see 7.6), as applicable, but reference to this separate document should be stated in the Test Plan.

6.2.7.8 Retesting and regression testing

Specifies the conditions under which retesting and regression testing will be performed. This could include a description of the estimated number of test cycles.

6.2.7.9 Suspension and resumption criteria

Specifies the criteria used to suspend and resume all or a portion of the testing activities in the Test Plan. Identifies who is responsible for suspending and resuming testing activities. Specifies the testing activities that may have to be repeated when testing is resumed.

6.2.7.10 Deviations from the Organizational Test Strategy

Records any Test Plan content that deviates from the Organizational Test Strategy. Identifies the authorities responsible for approving deviations, where applicable.

6.2.8 Testing activities and estimates

Identifies all necessary testing activities based on the test process to be used. The activity iteration strategy for the re-execution of test activities should be considered as well as any dependencies.

NOTE The testing activities could be described in terms of a work breakdown structure or activities on an activity board in agile projects.

EXAMPLE Activities that could be considered include those concerning retesting and regression testing.

Describes estimates for each of the identified testing activities to be performed as part of the testing activities covered by the test plan. Additionally, where appropriate, describes the allocated testing budget and cost estimates or references where that information can be found.

NOTE Budget and cost estimates could be located in the project plan.

6.2.9 Staffing

Describes the staffing requirements for the testing covered by this plan.

6.2.9.1 Roles, activities, and responsibilities

Provides an overview of the primary (they are the activity leader) and secondary (they are not the leader, but providing support) people filling the test-related roles and their corresponding responsibilities and authority for the testing activities. In addition, identifies those responsible for providing the test item(s). They may be participating either full- or part-time.

EXAMPLE The responsible parties could include the project manager, the test manager, the developers, the test analysts and executors, operations staff, user representatives, technical support staff, data administration staff, and quality support staff.

For each testing person, specify the period(s) when the person is required.

6.2.9.2 Hiring needs

Identifies specific requirements for additional testing staff that are necessary for the test project or test sub-process. Specifies when the staff are needed, if they should be temporary, full or part time, and the desired skill set. These may be defined by contract and business needs.

NOTE Staffing could be accomplished by internal transfer, external hiring, consultants, subcontractors, business partners, and/or outsourced resources.

6.2.9.3 Training needs

Specifies test training needs by skill level and identifies training options for providing the necessary skills for the staff needed.

EXAMPLE Training can take a variety of forms, including options such as traditional classroom training, self-paced computer-based training, training over the Internet, visiting the future user site, and mentoring by more knowledgeable staff members.

6.2.10 Schedule

Identifies test milestones defined in the project schedule and from the test strategy. Summarizes the overall schedule of the testing activities, identifying where activity results feed back to the development, organizational, and supporting processes. Specifies the schedule for each testing activity and test milestones based on the activity estimates, available resources, and other constraints.

EXAMPLE Supporting processes could be quality assurance and configuration management.

6.3 Test Status Report

6.3.1 Overview

The Test Status Report provides information about the status of the testing that is performed in a specific reporting period.

NOTE In an agile project, the Test Status Report might not be a written document. For example, its contents could be discussed at iteration meetings and supplemented by information stored on activity boards and burn-down charts.

Annex A.2.5 provides an outline of the Test Status Report, while Annexes G.1 and G.2 provide examples that demonstrate how Test Status Reports could be developed for two different example projects.

The contents of the Test Status Report include:

6.3.2 Document specific information

6.3.2.1 Overview

This information identifies the document and describes its origins and history.

NOTE The information could be placed on an early page in a document, or in a central place if the contents are kept in electronic form, e.g. in a database.

6.3.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

6.3.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

6.3.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

6.3.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change could be document author, project manager, system owner.

6.3.3 Introduction

Provides explanatory information about the context and structure of the document.

6.3.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

6.3.3.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into "external" references that are imposed from outside the organization and "internal" references that are imposed from within the organization.

6.3.3.3 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

6.3.4 Test status

Provides information on the status of the testing for the reporting period.

6.3.4.1 Reporting period

Specifies the time period covered by the report.

6.3.4.2 Progress against Test Plan

Describes the progress that has been made against the Test Plan. Any notable deviations from the plan should be highlighted, with explanations of the reasons for deviation, description of any remedial actions, an account of the effects, and consideration of the implications with regard to planned project objectives.

6.3.4.3 Factors blocking progress

Identifies those factors that impeded progress during the reporting period and the corresponding solutions that were implemented to remove them. Outstanding (unsolved) issues still impeding progress should be recorded and possible solutions identified.

6.3.4.4 Test measures

Presents the collated test measures related to the end of the reporting period.

EXAMPLE This could include measures on test cases, defects, incidents, test coverage, activity progress and resource consumption.

6.3.4.5 New and changed risks

Lists the new risks that have been identified as a result of monitoring and controlling testing as well as changes to existing risks during the reporting period.

6.3.4.6 Planned testing

Describes the planned testing for the next reporting period.

6.4 Test Completion Report

6.4.1 Overview

The Test Completion Report provides a summary of the testing that was performed. This may be for the project/programme as a whole or for the particular test sub-process.

Annex A.2.6 provides an outline of the Test Completion Report, while Annexes H.1 and H.2 provide examples that demonstrate how Test Completion Reports could be developed for two different example projects.

The contents of the Test Completion Report include:

6.4.2 Document specific information

6.4.2.1 Overview

This information identifies the document and describes its origins and history.

NOTE The information could be placed on an early page in a document, or in a central place if the contents are kept in electronic form, e.g. in a database.

6.4.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

6.4.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

6.4.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

6.4.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change could be document author, project manager, system owner.

6.4.3 Introduction

Provides explanatory information about the context and structure of the document.

6.4.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

6.4.3.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into "external" references that are imposed from outside the organization and "internal" references that are imposed from within the organization.

6.4.3.3 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

6.4.4 Testing performed

Provides a description of the testing performed.

6.4.4.1 Summary of testing performed

Summarizes the testing performed across the project and/or in the test sub-processes that are in scope for this report.

Provides details on what was tested, and describes any constraints on how the testing was performed.

EXAMPLE This could include restrictions on test environment availability.

6.4.4.2 Deviations from planned testing

Describes deviations from the planned testing, if any. This section may also reference the section on residual risks for any risks that the deviations pose to the testing and their relevant risk treatments.

6.4.4.3 Test completion evaluation

Describes the extent to which the testing met the specified test completion criteria, and where necessary, explains why the criteria were not met. This section may also reference the section on residual risks for any risks that any incomplete completion criteria pose to the testing and their relevant risk treatments.

6.4.4.4 Factors that blocked progress

Identifies those factors that impeded progress and the corresponding solutions that were implemented to remove them.

6.4.4.5 Test measures

Presents the collated test measures.

EXAMPLE This could include measures for test cases, defects, incidents, test coverage, activity progress, and resource consumption.

6.4.4.6 Residual risks

Lists the risks that are untreated at the end of the testing; this may be risks that have not been fully treated by the test and /or any new risks identified as a result of the final monitoring and closure of the test.

6.4.4.7 Test deliverables

Lists all the test deliverables produced as a result of the test effort and their location.

EXAMPLE This could include the Test Plan, Test Case Specifications and Test Procedure Specifications.

6.4.4.8 Reusable test assets

Lists all the reusable test assets and their location.

EXAMPLE This could include test procedures and test data that were produced as a result of the test effort.

6.4.4.9 Lessons learned

Describes the results of the lessons learned meeting.

7 Dynamic Test Processes Documentation

7.1 Overview

The documents developed in the dynamic test processes comprise the following types:

- Test Specification, divided into:
  - Test Design Specification;
  - Test Case Specification;
  - Test Procedure Specification.

NOTE These could be separate documents, could appear as chapters in a test specification document, or could be in the form of charters depending on the size and nature of the test project.

  - Test Data Requirements;
  - Test Environment Requirements;
  - Test Data Readiness Report;
  - Test Environment Readiness Report;
  - Test Execution Documentation, divided into:
  - Actual Results;
  - Test Results;
  - Test Execution Log.
  - Incident Report.

The full templates with explanatory text for the documents are found below. Annex A provides an abbreviated overview of each document. Annexes I to S provide examples of the dynamic test process documentation for example organizations.

NOTE There are many documentation styles and names, e.g. in agile, session sheets and charters with test ideas. It is expected that in the tailoring defined in clause 2.2, other names might be substituted for the names in this part. A mapping could be produced. In the annexes to this standard, examples for two different project types are presented with options for tailoring of names. The annexes are not inclusive of all document names, formats, and test methodologies, but intended to show some possible options.

7.2 Test Design Specification

7.2.1 Overview

The Test Design Specification identifies the features to be tested, and the test conditions derived from the test basis for each of the features as the first step towards the definition of test cases and test procedures to be executed.

Annex A.2.7 provides an outline of the Test Design Specification, while Annexes I.1 and I.2 provide examples that demonstrate how Test Design Specifications could be developed for two different example projects.

The contents of the Test Design Specification include:

7.2.2 Document specific information

7.2.2.1 Overview

This information identifies the document and describes its origins and history.

NOTE The information could be placed on an early page in a document, or in a central place if the contents are kept in electronic form, e.g. in a database.

7.2.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

7.2.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

7.2.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

7.2.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change could be document author, project manager, system owner.

7.2.3 Introduction

Provides explanatory information about the context and structure of the document.

7.2.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

7.2.3.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into “external” references that are imposed from outside the organization and “internal” references that are imposed from within the organization.

7.2.3.3 Notation convention(s)

Defines and explains any identification or numbering schemes needed for test sets and test conditions, if this is not defined elsewhere.

NOTE This could be found in a Configuration Management Plan.

7.2.3.4 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

7.2.4 Feature sets

7.2.4.1 Overview

A feature set is a logical grouping of the features to be tested for the test item(s), which are specified in the Test Plan. The features to be tested may be described by a single feature set or a number of feature sets, possibly arranged in a hierarchy. The feature sets may correspond directly to the architecture of the test item(s), or may be different if this facilitates a more efficient test. A feature set may also be a business process that consists of a series of features. Each feature set may be treated independently of other feature sets in the subsequent test design activities.

The feature sets may be described in lists or in tables in a document or using a tool.

EXAMPLE Feature sets could be stored in a database or a dedicated test tool.

The information for a feature set includes:

7.2.4.2 Unique identifier

Describes the unique identifier for the feature set so that it can be distinguished from all other feature sets. An automated tool may control the generation of the identifiers, or it may be done manually according to the appropriate notation scheme. The unique identifier is not to be changed during the lifetime of the feature set, because it is used for traceability purposes.

7.2.4.3 Objective

Identifies and briefly describes the special focus or objective for the feature set.

7.2.4.4 Priority

Defines the priority for the testing of this particular feature set, if needed.

7.2.4.5 Specific strategy

Specifies the implementation of the strategy for testing the feature set.

EXAMPLE This could include the use of specific test design techniques, defined in the corresponding Test Plan, if needed.

7.2.4.6 Traceability

Lists reference(s) to the associated features in the test basis.

NOTE The traceability could be documented in a Test Traceability Matrix or a tool.

EXAMPLE Features could be requirement(s) and/or design description(s).

7.2.5 Test conditions

7.2.5.1 Overview

Summarizes the test conditions for a feature set. A test condition is an individual item or event specified in the test basis that could be verified by one or more test cases.

NOTE A test condition could simply be a reference to a requirement (if the requirement is expressed in a verifiable way, i.e. if it includes an identifiable acceptance criterion), or to a design description. A test condition could also be a rewording of a requirement, a set of requirements, or a design description created for the purpose of testing, e.g. summarising a number of requirements in a decision table model or as a state model.

This section in the Test Design Specification could be formatted to list test conditions under corresponding feature sets.

NOTE The test conditions could be described in lists or in tables in a document or using a tool, e.g. a database or a dedicated test tool. Test conditions are not always formally documented, as they could be seen as a first draft of test coverage items and/or test cases.

The information for a test condition includes:

7.2.5.2 Unique identifier

Describes the unique identifier for the test condition so that it can be distinguished from all other test conditions. An automated tool may control the generation of the identifiers, or it may be done manually according to the appropriate notation scheme. The unique identifier is not to be changed during the lifetime of the test condition, because it is used for traceability purposes.

When the number or volatility of test conditions is so high that these requirements for unique identifiers become impractical, they may be replaced by the use of other means, generally based on automated tools, to establish traceability between test cases and test conditions.

7.2.5.3 Description

Describes the test condition, i.e. what can be tested. This may be written in natural language, and/or expressed as tabular or graphical models, as appropriate. It may simply reference the requirement that serves as the test condition.

7.2.5.4 Priority

Defines the priority for the testing of this particular test condition within the feature set. The higher priory test conditions will be tested earlier and more extensively than the lower priority test conditions.

7.2.5.5 Traceability

Describes traceability to the feature set or lists reference(s) to the associated requirement(s) and/or design description(s) in the test basis. This could be documented in a Test Traceability Matrix.

7.3 Test Case Specification

7.3.1 Overview

The Test Case Specification identifies the test coverage items and the corresponding test cases derived from the test basis for one or more feature sets.

Annex A.2.8 provides an outline of the Test Case Specification, while Annexes J.1 and J.2 provide examples that demonstrate how Test Case Specification could be developed for two different example projects.

The contents of the Test Case Specification include:

7.3.2 Document specific information

7.3.2.1 Overview

This information identifies the document and describes its origins and history.

NOTE The information could be placed on an early page in a document, or in a central place if the contents are kept in electronic form, e.g. in a database.

7.3.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

7.3.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

7.3.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

7.3.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change may be document author, project manager, system owner.

7.3.3 Introduction

Provides explanatory information about the context and structure of the document.

7.3.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

7.3.3.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into “external” references that are imposed from outside the organization and “internal” references that are imposed from within the organization.

7.3.3.3 Notation convention(s)

Defines and explains any identification or numbering schemes needed for test coverage items and test cases, if this is not defined elsewhere.

NOTE This could be found in a Configuration Management Plan.

7.3.3.4 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

7.3.4 Test coverage items

7.3.4.1 Overview

Summarizes the test coverage items for the test conditions. Test coverage items are derived by applying a test design technique to the test condition.

EXAMPLE Equivalence partitioning will identify test coverage items in terms of valid and invalid equivalence partitions.

This section in the Test Case Specification could be formatted to list test coverage items under corresponding feature sets and/or test conditions.

NOTE The test coverage items could be described in lists or in tables in a document or using a tool, e.g. a database or a dedicated test tool. Test coverage items are not always formally documented, as they could be seen as a draft of test cases.

The information for a test coverage item includes:

7.3.4.2 Unique identifier

Describes the unique identifier for the test coverage item so that it can be distinguished from all other test coverage items. An automated tool may control the generation of the identifiers, or it may be done manually according to the appropriate notation scheme. The unique identifier is not to be changed during the lifetime of the test coverage item, because it is used for traceability purposes.

When the number or volatility of test conditions is so high that these requirements for unique identifiers become impractical, they may be replaced by the use of other means, generally based on automated tools, to establish traceability between test cases and test conditions.

7.3.4.3 Description

Describes the test coverage item, i.e. what is expected to be covered by a test case according to the test design technique that was used during its derivation. It may also include additional information about the coverage item.

EXAMPLE Whether or not it is a valid or invalid equivalence partition.

7.3.4.4 Priority

Defines the priority for the testing of this particular test coverage item within the test condition, if needed. The higher priority test coverage items will be run before the lower priority test coverage items.

7.3.4.5 Traceability

Describes traceability to the test condition or feature set that the test coverage item belongs to or lists reference(s) to the associated test basis. This may be documented in a Test Traceability Matrix.

EXAMPLE Test basis could be requirements or design.

7.3.5 Test cases

7.3.5.1 Overview

Defines the test cases derived from the test coverage items. A test case specifies how one or more test coverage item(s) are exercised to help determine whether or not that part of the test item has been implemented correctly.

The number of test cases derived from test coverage items will depend on the test coverage criterion defined in the Test Plan.

This section in the Test Case Specification could be formatted to list test cases under corresponding feature sets and/or test conditions.

NOTE The test cases could be described in lists or in tables in a document or using a tool, e.g. a database or a dedicated test tool.

The information for a test case includes:

7.3.5.2 Unique identifier

Describes the unique identifier for the test case so that it can be distinguished from all other test cases. An automated tool may control the generation of the identifiers or it may be done manually according to the appropriate notation scheme. The unique identifier is not to be changed during the lifetime of the test case item, because it is used for traceability purposes.

7.3.5.3 Objective

Identifies and briefly describes the special focus or objective of the test case. This is typically in the form of a title.

7.3.5.4 Priority

Defines the priority for the testing of this particular test case, if needed. The higher priority test cases will be run before the lower priority test cases.

7.3.5.5 Traceability

Describes traceability to the test coverage item that the test case exercises or lists reference(s) to the associated requirement(s) and/or design description(s) in the test basis. This may be documented in a Test Traceability Matrix.

7.3.5.6 Preconditions

Describes the required state of the test environment and any special constraints pertaining to the execution of the test case.

EXAMPLE The state the test item must be in before execution could start, including existence of specific test data and the currently active form or screen.

This could be described explicitly or it could include references to other test cases, whose execution will set the preconditions.

The environment needed may be described collectively for one or more feature sets, or it may not be described in this specification if the description in the Test Plan is sufficient.

7.3.5.7 Inputs

Specifies each action required to bring the test item into a state where the expected result can be compared to the actual results. The detail of the descriptions should be tailored to fit the knowledge of the test executors.

NOTE This could require provision of input data and/or events, e.g. button clicks, to the test item. Some of the input data may be specified by value, while others may be specified by name. Constant tables, transaction files, databases, files, terminal messages, memory resident areas, and values passed by the operating system must be considered.

All required relationships between input events must be described.

EXAMPLE A relationship could be timing.

The actions may be numbered within the test case, if needed.

7.3.5.8 Expected results

Specifies the expected outputs and behaviour required of the test item in response to the inputs that are given to the test item when it is in its precondition state. Provides the expected value(s) (with tolerances where appropriate) for each required output.

EXAMPLE Behaviour required of the test item could be response time.

The actions required to compare the expected results to the actual results may also be specified. For instance, examining the output in a field on a form that is not active when the input is provided, waiting for a batch job to run and a report to be printed out and examined, or closing down the test item and restarting it to examine stored data.

7.3.5.9 Actual results and test result

The description of a test case may include placeholders to record actual results and/or test result during execution of the test case. Alternatively, these may be recorded in the Test Procedure Specification (see clause 7.4), or separately in the Actual Results (see clause 7.9) and/or Test Result (see clause 7.10).

7.4 Test Procedure Specification

7.4.1 Overview

The Test Procedure Specification describes the test cases in the selected test sets in execution order, along with any associated actions that may be required to set up the initial preconditions and any post execution wrap up activities.

NOTE The test procedures could be described in lists or in tables in a document or using a tool, e.g. a database or a dedicated test tool.

Annex A.2.9 provides an outline of the Test Procedure Specification, while Annexes K.1 and K.2 provide examples that demonstrate how Test Procedure Specifications could be developed for two different example projects.

The contents of the Test Procedure Specification include:

7.4.2 Document specific information

7.4.2.1 Overview

This information identifies the document and describes its origins and history.

NOTE The information could be placed on an early page in a document, or in a central place if the contents are kept in electronic form, e.g. in a database.

7.4.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

7.4.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

7.4.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

7.4.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change could be document author, project manager, system owner.

7.4.3 Introduction

Provides explanatory information about the context and structure of the document.

7.4.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

7.4.3.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into “external” references that are imposed from outside the organization and “internal” references that are imposed from within the organization.

7.4.3.3 Notation convention(s)

Defines and explains any identification or numbering schemes needed for test sets and test procedures, if this is not defined elsewhere.

NOTE This could be found in a configuration management plan.

7.4.3.4 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

7.4.4 Test sets

7.4.4.1 Overview

Describes the assembly of test cases into test sets for the purpose of testing a specific test objective. The test sets will typically reflect the feature sets, but they may contain test cases for a number of feature sets. Test cases for a test set may be selected based on the identified risks, test basis, retesting and/or regression testing.

NOTE The test sets could be described in lists or in tables in a document or using a tool, e.g. a database or a dedicated test tool. Test sets are not always formally documented, as they could be seen as a drafting step for the creation of test procedures.

The information for a test set includes:

7.4.4.2 Unique identifier

Describes the unique identifier for the test set so that it can be distinguished from all other test sets. An automated tool may control the generation of the identifiers or it may be done manually according to the appropriate notation scheme. The unique identifier is not to be changed during the lifetime of the test set, because it is used for traceability purposes.

7.4.4.3 Objective

Identifies and briefly describes the special focus or objective for the test set.

EXAMPLE “This test set is targeted at retesting corrections regarding incidents IN301 and IN56”.

7.4.4.4 Priority

Defines the priority for the testing of this particular test set, if needed.

7.4.4.5 Contents (Traceability)

Summarizes the contents of the test set. This will usually be a list of the unique identifiers of the selected test cases.

7.4.5 Test procedures

7.4.5.1 Overview

Describes the test procedures that have been derived for the test sets. A test procedure specifies the order in which the test cases in the corresponding test set should be executed according to dependencies described by pre-conditions and post-conditions and other testing requirements.

NOTE The test procedures could be described in lists or in tables in a document or using a tool, e.g. a database or a dedicated test tool.

The information for a test procedure includes:

7.4.5.2 Unique identifier

Describes the unique identifier for the test procedure so that it can be distinguished from all other test procedures. An automated tool may control the generation of the identifiers or it may be done manually according to the appropriate notation scheme. The unique identifier is not to be changed during the lifetime of the test procedure, because it is used for traceability purposes.

7.4.5.3 Objective

Identifies and briefly describes the special focus or objective of the test procedure. This will be identical to the objective of the corresponding test set.

7.4.5.4 Priority

Defines the priority for this particular test procedure, if needed. This will be identical to the priority of the corresponding test set.

7.4.5.5 Start up

Describes the necessary actions to prepare for execution of the test cases specified in the test procedure. This will typically be the actions to set up the preconditions for the first test case to be executed.

7.4.5.6 Test cases to be executed (Traceability)

Lists the test cases in the order in which they are to be executed. The test cases may be numbered sequentially within the test procedure. The degree to which the procedure can be varied may be defined.

This list may be a reference to the test cases, or it may be a copied list of the test cases.

If the execution of one or more test cases in the procedure does not set the preconditions for the following test case, actions to set the preconditions may be added between test cases.

The test procedure may include placeholders to record actual results and/or test results. Alternatively, actual results and/or test results may be recorded in the Actual Results document (see 7.9) and/or Test Result document (see clause 7.10).

7.4.5.7 Relationship to other procedures

Describes dependencies this test procedure may have on any other test procedures.

Examples of dependencies on other test procedures include that they are executed before this one, concurrently with this one, or subsequent to this one.

7.4.5.8 Stop and wrap up

Describes the actions necessary to bring execution to an orderly halt and the actions necessary after the execution of the procedure has been completed.

EXAMPLE Actions could be termination of logging or resetting of test database.

7.5 Test Data Requirements

7.5.1 Overview

The Test Data Requirements describe the properties of the test data needed to execute the test procedures defined in the Test Procedure Specification.

Annex A.2.10 provides an outline of the Test Data Requirements, while Annexes L.1 and L.2 provide examples that demonstrate how Test Data Requirements could be developed for two different example projects.

The contents of the Test Data Requirements include:

7.5.2 Document specific information

7.5.2.1 Overview

This information identifies the document and describes its origins and history.

NOTE The information could be placed on an early page in a document, or in a central place if the contents are kept in electronic form, e.g. in a database.

7.5.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

7.5.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

7.5.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

7.5.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change may be for example document author, project manager, system owner.

7.5.3 Introduction

Provides explanatory information about the context and structure of the document.

7.5.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

7.5.3.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into “external” references that are imposed from outside the organization and “internal” references that are imposed from within the organization.

7.5.3.3 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

7.5.4 Detailed test data requirements

7.5.4.1 Overview

Describes the data required to execute the test procedures defined in the Test Procedure Specification. It may also contain any requirements to sanitize data.

EXAMPLE Test data could describe simulated production data, such as customer data and user account data.

This data may be divided into elements reflecting the data structure of the test item.

EXAMPLE The data could be defined in a class diagram or an entity-relationship diagram.

The information for the test data requirements includes:

7.5.4.2 Unique identifier

Describes the unique identifier for the test data requirement so that it can be distinguished from all other test data requirements. An automated tool may control the generation of the identifiers or it may be done manually according to the appropriate notation scheme. The unique identifier is not to be changed during the lifetime of the test data requirement, because it is used for traceability purposes.

7.5.4.3 Description

Defines the specific name and required values or ranges of values for each test data element. It may also describe when data needs to be made anonymous or manipulated in other ways.

EXAMPLE “At least 10 customers must exist in the database with complete and correct CustomerID and all other mandatory customer information.”

7.5.4.4 Responsibility

Specifies who is responsible for making the test data available.

7.5.4.5 Period needed

Identifies when and for how long the test data is needed. Test data may be needed for a single undivided period or for several separate periods.

7.5.4.6 Resetting needs

Specifies if the test data needs to be reset during testing.

7.5.4.7 Archiving or disposal

Identifies when and how test data may be archived or disposed of after completion of the testing.

7.6 Test Environment Requirements

7.6.1 Overview

The Test Environment Requirements describe the properties of the test environment needed to execute the test procedures defined in the Test Procedure Specification. Where appropriate, this document may simply reference where the relevant information can be found.

EXAMPLE The information could be found in the Organizational Test Strategy, Test Plan or Test Specification.

Annex A.2.11 provides an outline of the Test Environment Requirements, while Annexes M.1 and M.2 provide examples that demonstrate how Test Environment Requirements could be developed for two different example projects.

The contents of the Test Environment Requirements document include:

7.6.2 Document specific information

7.6.2.1 Overview

This information identifies the document and describes its origins and history.

EXAMPLE The information could be placed on an early page in a document, or in a central place if the contents are kept in electronic form, e.g. in a database.

7.6.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

7.6.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

7.6.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

7.6.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change could be document author, project manager, system owner.

7.6.3 Introduction

Provides explanatory information about the context and structure of the document.

7.6.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

7.6.3.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into “external” references that are imposed from outside the organization and “internal” references that are imposed from within the organization.

7.6.3.3 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

7.6.4 Detailed test environment requirements

7.6.4.1 Overview

Identifies the environment items required to execute the test procedures defined in the Test Procedure Specification. This includes set-up before the execution of the test procedures, for execution of the test procedures, and for any activities after test execution.

EXAMPLE The environment items could be divided into the following types, although other types could be needed depending on the specific environment requirements:

- Hardware;
- Middleware;
- Software;
- Peripherals, e.g. printers;
- Communication means, e.g. web access;
- Tools;
- Security;
- Venue, e.g. size of rooms and background noise level;
- Accessories, e.g. special pre-printed paper forms.

NOTE The environment items could be grouped using other criteria, e.g. WindowsXP/Vista/Windows7 or different front ends connected to the PCs, if this is more appropriate. Descriptions of the specific configuration(s) where these items are to be used and/or reused could also be included.

In practise the test environment will usually not be a perfect representation of the operational environment and the detailed environment requirements should reflect the degree to which the testing environment needs to represent the operational environment.

The information for a test environment item includes:

7.6.4.2 Unique identifier

Describes the unique identifier for the environment item so that it can be distinguished from all other environment items. An automated tool may control the generation of the identifiers or it may be done manually according to the appropriate notation scheme. The unique identifier is not to be changed during the lifetime of the test environment item, because it is used for traceability purposes.

7.6.4.3 Description

Identifies the environment item in sufficient detail for it to be delivered as expected.

EXAMPLE This could include precisely named hardware or software in specific versions and specific configurations. It could also list required batch jobs that need to be run at certain points during testing to support the testing process.

7.6.4.4 Responsibility

Specifies who is responsible for making the environment item available.

7.6.4.5 Period needed

Identifies when and for how long the environment item is needed. An environment item may be needed for a single undivided period or for several separate periods.

7.7 Test Data Readiness Report

7.7.1 Overview

The Test Data Readiness Report describes the fulfilment of each test data requirement.

Annex A.2.12 provides an outline of the Test Data Readiness Report, while Annexes N.1 and N.2 provide examples that demonstrate how Test Data Readiness Reports could be developed for two different example projects.

The contents of the Test Data Readiness Report include:

7.7.2 Document specific information

7.7.2.1 Overview

This information identifies the document and describes its origins and history.

EXAMPLE The information could be placed on an early page in a document, or in a central place if the contents are kept in electronic form, e.g. in a database.

7.7.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

7.7.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

7.7.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

7.7.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change could be document author, project manager, system owner.

7.7.3 Introduction

Provides explanatory information about the context and structure of the document.

7.7.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

7.7.3.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into “external” references that are imposed from outside the organization and “internal” references that are imposed from within the organization.

7.7.3.3 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

7.7.4 Test data status

7.7.4.1 Overview

Provides a status for each test data requirement. This may be noted in a placeholder in the Test Data Requirements document.

The information for each data item includes:

7.7.4.2 Unique identifier

The unique identifier used in the Test Data Requirements document.

7.7.4.3 Description of status

Describes the status of the required test data item. The status may include a description of how the actual test data deviates from the requirements, e.g. in terms of values or volume.

7.8 Test Environment Readiness Report

7.8.1 Overview

The Test Environment Readiness Report describes the fulfilment of each test environment requirement.

Annex A.2.13 provides an outline of the Test Environment Readiness Report, while Annexes O.1 and O.2 provide examples that demonstrate how Test Environment Readiness Reports could be developed for two different example projects.

The contents of the Test Environment Readiness Report include:

7.8.2 Document specific information

7.8.2.1 Overview

This information identifies the document and describes its origins and history.

EXAMPLE The information could be placed on an early page in a document, or in a central place if the contents are kept in electronic form, e.g. in a database.

7.8.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

7.8.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

7.8.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

7.8.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change could be document author, project manager, system owner.

7.8.3 Introduction

Provides explanatory information about the context and structure of the document.

7.8.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

7.8.3.2 References

Lists referenced documents and identify repositories for system, software, and test information. The references may be separated into “external” references that are imposed from outside the organization and “internal” references that are imposed from within the organization.

7.8.3.3 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

7.8.4 Test environment readiness

7.8.4.1 Overview

Provides a fulfilment statement for each test environment requirement. This may be noted in a placeholder in the Test Environment Requirements document.

The information for each environment item includes:

7.8.4.2 Unique identifier

The unique identifier used in the Test Environment Requirements document.

7.8.4.3 Description of status

Describes the fulfilment of the required test environment item.

EXAMPLE The fulfilment statement could include a description of how the actual test environment deviates from the requirements, e.g. in terms of versions or configuration.

NOTE This section could also be used to record the availability of certain test environment items (e.g. other applications that are integrated with the test item) as these could have an impact on testing if they are unavailable.

7.9 Actual Results

The actual results are a record of the result of the execution of a test case in a test procedure. The actual results are compared to the expected results to determine the test result.

The actual results are not always formally recorded. Some types of systems (e.g. regulated safety critical) may be required to fully document the actual results, and some systems (e.g. those with high data integrity or reliability requirements) may choose to do full recording of actual results. The recording may be done by an automated tool during test execution.

Some test cases may include actions that provide results, which are not part of the actual results of executing the test case, but intermediate results. These may be recorded separately in the test log or with the actual results. In the latter case a clear distinction must be made between the actual results and the intermediate results.

When needed, the actual results are usually recorded directly in the test procedure in a placeholder reserved for this purpose. The actual results are therefore not usually considered as an independent document.

7.10 Test Result

The test result is a record of whether or not a specific test case execution has passed or failed, i.e. if the actual results correspond to the expected results or if deviations were observed, or if planned execution of the test case was not possible.

The test result for a test case is usually recorded directly in the test procedure in a placeholder reserved for this purpose. The test result is therefore not usually considered as an independent document.

EXAMPLE If the test case passes this could be marked with a tick mark or the like, and if the test case execution fails this could be marked with the number of the incident report raised as a result of the observation of the deviation. If the test case could not be executed this could be marked as "blocked"; in this case also record what the blockage is, if possible.

Sometimes this process is fully automated, with the tool comparing the actual results to expected results, and providing a report of which test cases passed, failed, or could not be executed.

NOTE Test Result is sometimes referred to as “Pass/Fail”.

7.11 Test Execution Log

7.11.1 Overview

Records details of the execution of one or more test procedures.

The test procedures may be described in lists or in tables in a document or produced by a tool, e.g. a database or a dedicated test tool.

Annex A.2.14 provides an outline of the Test Execution Log, while Annexes R.1 and R.2 provide examples that demonstrate how the Test Execution Log could be developed for two different example projects.

The contents of the Test Execution Log include:

7.11.2 Document specific information

7.11.2.1 Overview

This information identifies the document and describes its origins and history.

EXAMPLE The information could be placed on an early page in a document, or in a central place if the contents are kept in electronic form, e.g. in a database.

7.11.2.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

7.11.2.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s) and the tester(s) if these are not the same.

7.11.2.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

7.11.2.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change could be document author, project manager, system owner.

7.11.3 Introduction

Provides explanatory information about the context and structure of the document.

7.11.3.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

7.11.3.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into “external” references that are imposed from outside the organization and “internal” references that are imposed from within the organization.

7.11.3.3 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

7.11.4 Events

7.11.4.1 Overview

Lists the significant events encountered during the execution of one or more test procedures.

EXAMPLE The first event could be the start of the test execution session, and the last event could be the final closing of the test execution session.

Further examples of events to record include:

- A sudden drop in performance of the computer on which the test is being executed;
- A failure making further execution of the test impossible;
- A disruption to the test environment causing the actual results to be unreliable.

The information for each event recorded in the Test Execution Log includes:

7.11.4.2 Unique identifier

Defines the sequential number of the entry in the Test Execution Log.

7.11.4.3 Time

Defines the precise time, including date if necessary, when the event was encountered.

7.11.4.4 Description

Describes what happened. This may include a reference to the test procedure and test case being executed when the event was encountered, if relevant.

7.11.4.5 Impact

Describes the impact on test execution and/or the actual result, if relevant.

7.12 Test Incident Reporting

7.12.1 Overview

A test incident is any issue that is noticed during testing that requires action(s) to be documented. Test incidents are recorded in incident reports. There will be one incident report for each unique incident (incident reports may also be known as defect reports, bug reports, fault reports, etc.).

The incident reports may be documented in lists or in tables in a document or using a tool, e.g. a database or a dedicated bug-tracking tool.

The format of an incident report may be defined elsewhere in an organization, for example as part of the Incident Management Processes, in which case that definition should be used.

7.12.2 Incident Report

An Incident Report in this context documents an incident recognized during testing.

NOTE 1 Incidents could occur, and be reported, in other contexts, e.g., ambiguities in a business requirements specification discovered during software design, or a software failure occurring during production.

NOTE 2 The information given here is only the information needed when an incident report is first raised. More information could be added to the incident report as it passes through the wider incident management process.

Annex A.2.15 provides an outline of the Incident Report, while Annexes S.1 and S.2 provide examples that demonstrate how an Incident Report could be developed for two different example projects.

The contents of the Incident Report include:

7.12.3 Document specific information

7.12.3.1 Overview

This information identifies the document and describes its origins and history.

EXAMPLE The information could be placed on an early page in a document, or in a central place if the contents are kept in electronic form, e.g. in a database.

7.12.3.2 Unique identification of document

Uniquely identifies a version of the document.

EXAMPLE The unique identifier could include the title of the document, the date of issue, version, and/or document status (e.g. draft, reviewed, corrected, final).

7.12.3.3 Issuing organization

Specifies the organization responsible for preparing and releasing the document. It may also include the author(s).

7.12.3.4 Approval authority

Identifies the designated person(s) who have the responsibility for reviewing and signing off on the document (possibly electronically). It may also include the reviewers and pertinent managers.

7.12.3.5 Change history

Includes a log of all of the changes that have occurred to the document since its inception.

EXAMPLE 1 This could include a list including the present version of the document and any predecessor documents containing the unique identification of each document, description of document changes with respect to the previous document in the list, reason for changes, and the name and role of the person making the changes.

EXAMPLE 2 Reasons for changes could include audit comments, team review, and system changes, and person making the change could be document author, project manager, system owner.

7.12.4 Introduction

Provides explanatory information about the context and structure of the document.

7.12.4.1 Scope

Identifies the extent of the coverage of the subject area by the document, and describes any inclusions, exclusions, assumptions and/or limitations.

7.12.4.2 References

Lists referenced documents and identifies repositories for system, software, and test information. The references may be separated into “external” references that are imposed from outside the organization and “internal” references that are imposed from within the organization.

7.12.4.3 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the document.

NOTE This section could be an annex, or it could refer to another document providing a general glossary. All or part of the glossary and/or acronym list could be online, as a separate testing specific glossary or incorporated in a larger organizational glossary (including more terms than just those that are testing related).

7.12.5 Incident details

The information for the incident, when it is first recognized and reported, includes:

7.12.5.1 Timing information

Records the date (and possibly also the time) when the incident was first observed.

7.12.5.2 Originator

Specifies the name(s) and title(s) of the individual(s) who identified the incident.

7.12.5.3 Context

Identifies the context in which the incident was observed.

EXAMPLE This could include:

- Configuration Item, including its unique identification, in which the incident was observed. In a testing context the item will usually be the test item, but it might be another configuration item, for example the Test Specification;
- The Test Procedure and Test Case, including their unique identifiers, being executed when the incident was observed;
- Any relevant information about the test environment and/or test data not included in other documents and considered especially significant by the tester;
- The test process or sub-process in which the incident was observed.

7.12.5.4 Description of the incident

Provides a detailed description of the incident. Indicates if the incident is reproducible, and, if so, provides enough information to reproduce it.

Related information and observations that may help to isolate and correct the cause of the incident may be included.

The description may also reference the location(s) where additional evidence or supporting information can be found to aid in diagnosis of the incident.

EXAMPLE Such evidence could include screenshots, system logs and output files.

7.12.5.5 Originator's assessment of severity

Indicates, from the originator's point of view, the depth and breadth of the impact this incident will have on technical and business issues. This may include an estimate of the time and effort to fix the associated defect.

EXAMPLE Technical and business issues could be the user's ability to perform tasks and system operations.

Also identifies the existence of any known workarounds.

7.12.5.6 Originator's assessment of priority

Provides an evaluation of the urgency for the repair. Most organizations have from three to five categories.

EXAMPLE A categorization scheme could be that the most serious category, e.g. "Fix now", means that the product is unusable, and the least serious, e.g. "Fix in support", is a cosmetic incident.

7.12.5.7 Risk

Provides information on the introduction of new risks or changes to the status of existing risks, where applicable.

7.12.5.8 Status of the incident

Identifies the current status of the incident, which will be "Open" or similar in this context.

NOTE A common sequence for incidents as they progress through their life cycles could be: "Open", "Approved for resolution", "Assigned for resolution", "Fixed", "Retested with the fix confirmed", and "Closed". Other possible status values could be "Rejected" or "Withdrawn".

Annex A
(informative)

Overview and Outlines of Documents

A.1 Overview

Figure A.1 shows that context for test documentation is set by the Organizational Test Policy. The Dynamic Test documentation is developed within the context of the test management documentation for a particular project.

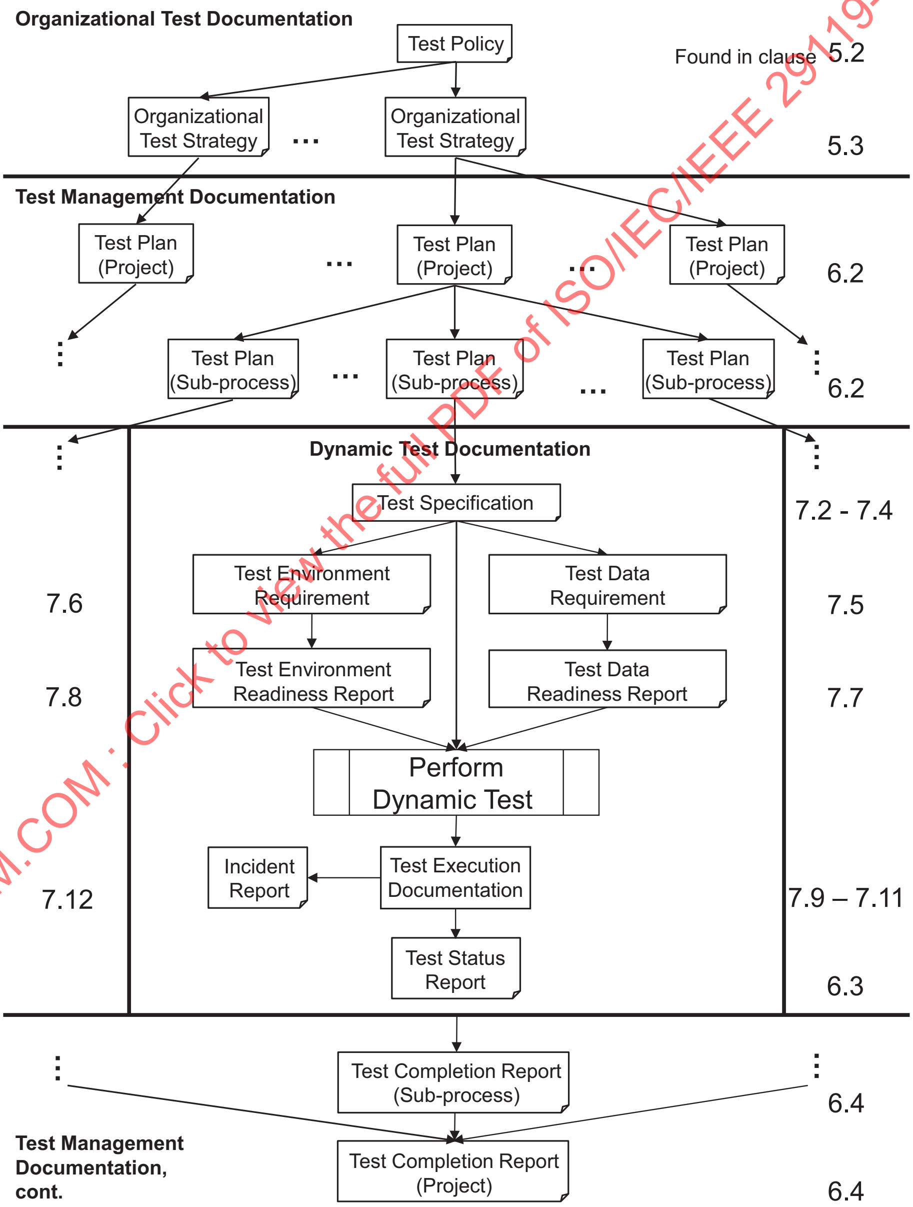

Organizational Test Documentation
Test Policy
Found in clause
5.2
Organizational Test Strategy
...
Organizational Test Strategy
5.3
Test Management Documentation
Test Plan (Project)
...
Test Plan (Project)
...
Test Plan (Project)
6.2
Test Plan (Sub-process)
...
Test Plan (Sub-process)
...
Test Plan (Sub-process)
6.2
Dynamic Test Documentation
Test Specification
7.2 - 7.4
7.6
Test Environment Requirement
Test Data Requirement
7.5
7.8
Test Environment Readiness Report
Test Data Readiness Report
7.7
Perform Dynamic Test
Incident Report
Test Execution Documentation
7.12
7.9 – 7.11
Test Status Report
6.3
Test Completion Report (Sub-process)
6.4
Test Management Documentation, cont.
Test Completion Report (Project)
6.4

Figure A.1 — The hierarchy of test documentation

Figure A.2 shows the hierarchy between the contents of the documents produced in completing the *Test Design and Implementation* process outlined in ISO/IEC/IEEE 29119-2.

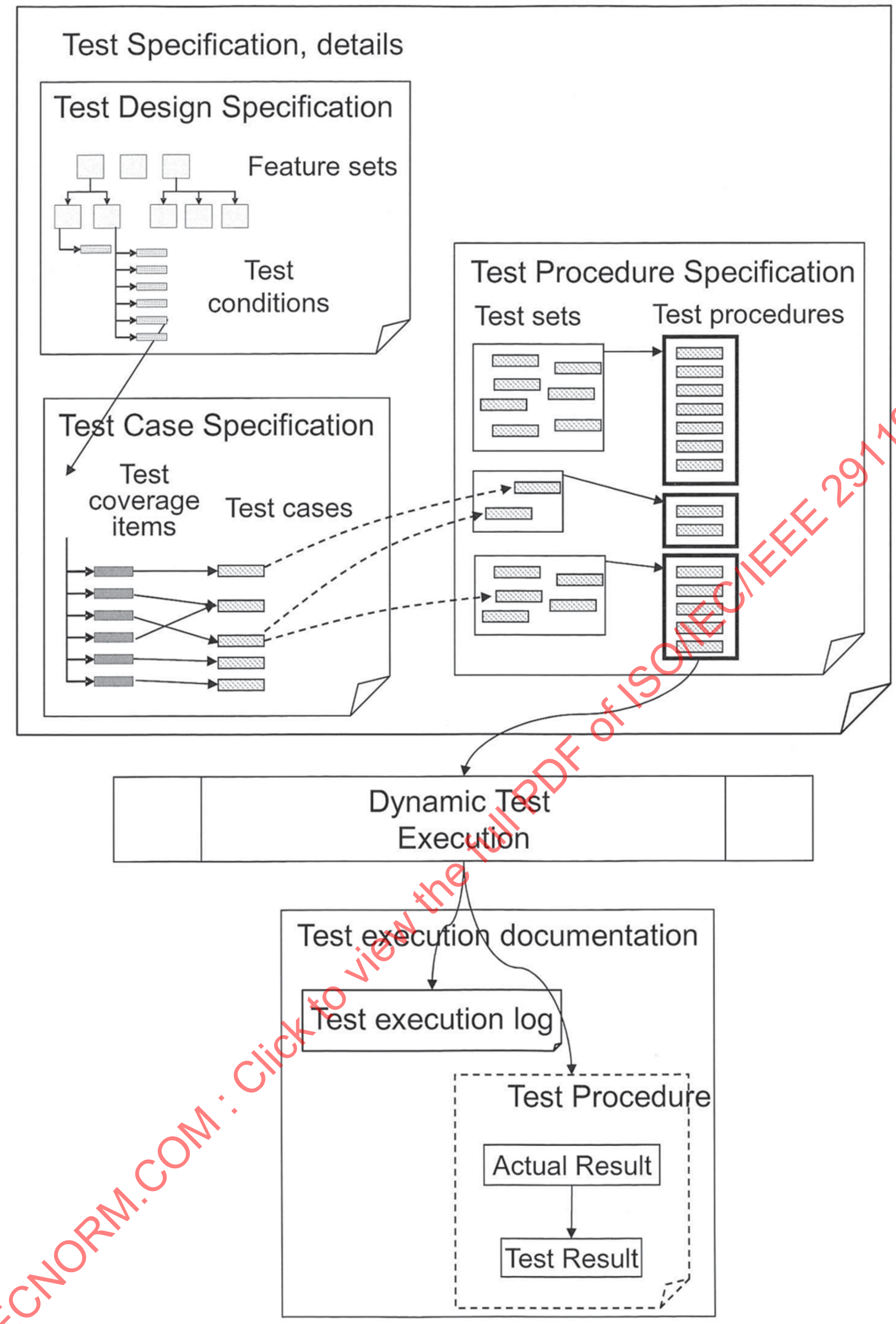


Figure A.2 — The hierarchy of test design and implementation documentation

A.2 Document Outlines

A.2.1 Overview

The outlines of the contents of each of the defined documents are provided below.

All documents include the following:

Document specific information:

- i) Unique identification of document
- ii) Issuing organization
- iii) Approval authority
- iv) Change history

Introduction:

- i) Scope
- ii) References
- iii) Glossary

The position of this information in the documents could vary from organization to organization and within an organization from document to document.

The specific document contents are usually placed after the Introduction.

A.2.2 Organizational Test Policy

The outline of the Test Policy specific information is:

a) Test policy statements:

- i) Objectives of testing
- ii) Test process
- iii) Test organization structure
- iv) Tester training
- v) Tester ethics
- vi) Standards
- vii) Other relevant policies
- viii) Measuring the value of testing
- ix) Test asset archiving and reuse
- x) Test process improvement

A.2.3 Organizational Test Strategy

The outline of the Organizational Test Strategy specific information is:

a) Project-wide organizational test strategy statements:

- i) Generic risk management
- ii) Test selection and prioritization

- iii) Test documentation and reporting
- iv) Test automation and tools
- v) Configuration management of test work products
- vi) Incident management
- vii) Test sub-processes

b) Test sub-process-specific organizational test strategy statements:

- i) Entry and exit criteria
- ii) Test completion criteria
- iii) Test documentation and reporting
- iv) Degree of independence
- v) Test design techniques
- vi) Test environment
- vii) Metrics to be collected
- viii) Retesting and regression testing

A.2.4 Test Plan

The outline of the Test Plan specific information is:

a) Context of the testing:

- i) Project/Test sub-process
- ii) Test item(s)
- iii) Test scope
- iv) Assumptions and constraints
- v) Stakeholders

b) Testing communication

c) Risk register:

- i) Product risks
- ii) Project risks

d) Test strategy:

- i) Test sub-processes
- ii) Test deliverables

iii) Test design techniques

iv) Test completion criteria

v) Metrics to be collected

vi) Test data requirements

vii) Test environment requirements

xi) Retesting and regression testing

xii) Suspension and resumption criteria

xiii) Deviations from the Organizational Test Strategy

e) Testing activities and estimates

f) Staffing:

i) Roles, activities, and responsibilities

ii) Hiring needs

iii) Training needs

g) Schedule

A.2.5 Test Status Report

The outline of the Test Status Report specific information is:

a) Test status:

i) Reporting period

ii) Progress against Test Plan

iii) Factors blocking progress

iv) Test measures

v) New and changed risks

vi) Planned testing

A.2.6 Test Completion Report

The outline of the Test Completion Report specific information is:

a) Testing performed:

i) Summary of testing performed

ii) Deviations from planned testing

iii) Test completion evaluation

iv) Factors that blocked progress

v) Test measures

vi) Residual risks

vii) Test deliverables

viii) Reusable test assets

ix) Lessons learned

A.2.7 Test Design Specification

The outline of the Test Design Specification specific information is:

a) Feature sets:

i) Unique identifier

ii) Objective

iii) Priority

iv) Specific strategy

v) Traceability

b) Test conditions:

i) Unique identifier

ii) Description

iii) Priority

iv) Traceability

A.2.8 Test Case Specification

The outline of the Test Case Specification specific information is:

a) Test coverage items:

i) Unique identifier

ii) Description

iii) Priority

iv) Traceability

b) Test cases:

i) Unique identifier

ii) Objective

iii) Priority

iv) Traceability

v) Preconditions

vi) Inputs

vii) Expected results

viii) Actual results and test result

A.2.9 Test Procedure Specification

The outline of the Test Procedure Specification specific information is:

a) Test sets:

   i) Unique identifier

   ii) Objective

   iii) Priority

   iv) Contents (Traceability)

b) Test procedures:

   i) Unique identifier

   ii) Objective

   iii) Priority

   iv) Start up

   v) Test cases to be executed (Traceability)

   vi) Relationship to other procedures

   vii) Stop and wrap up

A.2.10 Test Data Requirements

The outline of the Test Data Requirements specific information is:

a) Detailed test data requirements:

   i) Unique identifier

   ii) Description

   iii) Responsibility

   iv) Period needed

   v) Resetting needs

   vi) Archiving or disposal

A.2.11 Test Environment Requirements

The outline of the Test Environment Requirements specific information is:

a) Detailed test environment requirements:

   i) Unique identifier

   ii) Description

   iii) Responsibility

   iv) Period needed

A.2.12 Test Data Readiness Report

The outline of the Test Data Readiness Report specific information is:

a) Test data status:

   i) Unique identifier

   ii) Description of status

A.2.13 Test Environment Readiness Report

The outline of the Test Environment Readiness Report specific information is:

a) Test environment status:

   i) Unique identifier

   ii) Description of status

A.2.14 Test Execution Log

The outline of the Test Execution Log specific information is:

a) Events:

   i) Unique identifier

   ii) Time

   iii) Description

   iv) Impact

A.2.15 Incident Report

The outline of the Incident Report (recognition state) specific information is:

a) Incident details:

   i) Timing information

   ii) Originator

iii) Context

iv) Description of the incident

v) Originator’s assessment of severity

vi) Originator’s assessment of priority

vii) Risk

viii) Status of the incident

Annex B
(informative)

ISO/IEC/IEEE 29119-2 Normative Requirements Mapped to ISO/IEC/IEEE 29119-3 Information Items

B.1 Mapping

This informative annex explains at a high level how ISO/IEC/IEEE 29119-2 activities map to the information items in the documentation templates defined in ISO/IEC/IEEE 29119-3. For users of ISO/IEC/IEEE 29119-3 that do not use ISO/IEC/IEEE 29119-2, this annex is optional.

The table below summarizes the normative requirements for the clauses in ISO/IEC/IEEE 29119-2 where the creation of the ISO/IEC/IEEE 29119-3 information item is described.

Table B.1 — Summary of normative requirements from ISO/IEC/IEEE 29119-2 where an information item from ISO/IEC/IEEE 29119-3 is described

ISO/IE/IEEE 29119-3 Information Items	Normative requirement
B.1.1 Organizational Test Policy	
a) Test policy statements:	
i) Objectives of testing	Shall
ii) Test process	May
iii) Test organization structure	May
iv) Tester training	May
v) Tester ethics	May
vi) Standards	May
vii) Other relevant policies	May
viii) Measuring the value of testing	May
ix) Test asset archiving and reuse	May
x) Test process improvement	May
B.1.2 Organizational Test Strategy	
a) Project-wide organizational test strategy statements:	
i) Generic risk management	Shall
ii) Test selection and prioritization	Shall
iii) Test documentation and reporting	May
iv) Test automation and tools	May
v) Configuration management of test work products	May
vi) Incident management	May
vii) Test sub-processes	May
b) Test sub-process-specific organizational test strategy statements:	
i) Entry and exit criteria	May
ii) Test completion criteria	May
iii) Test documentation and reporting	May

ISO/IE/IEEE 29119-3 Information Items	Normative requirement
iv) Degree of independence	May
v) Test design techniques	May
vi) Test environment	May
vii) Metrics to be collected	May
viii) Retesting and regression testing	May
B.1.3 Test Plan	Shall
a) Context of the testing:	Shall
i) Project(s)/Test sub-process(es)	Shall
ii) Test item(s)	Shall
iii) Test scope	Shall
iv) Assumptions and constraints	Should
v) Stakeholders	Should
b) Testing communication	Should
c) Risk register:	Shall
i) Product risks	Shall
ii) Project risks	Shall
d) Test strategy:	Shall
i) Test sub-processes	Shall
ii) Test deliverables	Shall
iii) Test design techniques	Shall
iv) Test completion criteria	Shall
v) Metrics to be collected	Shall
vi) Test data requirements	Shall
vii) Test environment requirements	Shall
viii) Retesting and regression testing	Shall
ix) Suspension and resumption criteria	Shall
x) Deviations from the Organizational Test Strategy	Should
e) Testing activities and estimates	Shall
f) Staffing:	Should
i) Roles, activities, and responsibilities	Should
ii) Hiring needs	Should
iii) Training needs	Should
g) Schedule	Shall
B.1.4 Test Status Report	Shall
a) Test status:	Shall
i) Reporting period	Shall
ii) Progress against Test Plan	Shall
iii) Factors blocking progress	Shall
iv) Test measures	Shall
v) New and changed risks	Shall

ISO/IE/IEEE 29119-3 Information Items	Normative requirement
B.1.5 Test Completion Report	Shall
a) Testing performed:	Shall
i) Summary of testing performed	Shall
ii) Deviations from planned testing	Shall
iii) Test completion evaluation	Shall
iv) Factors that blocked progress	Shall
v) Test measures	Shall
vi) Residual risks	Shall
vii) Test deliverables	Shall
viii) Reusable test assets	Should
ix) Lessons learned	Shall
B.1.6 Test Design Specification	Shall
a) Feature sets:	Shall
i) Unique identifier	Shall
ii) Objective	Shall
iii) Priority	Shall
iv) Extract of strategy	Shall
v) Traceability	Shall
b) Test conditions:	Shall
i) Unique identifier	Shall
ii) Description	Shall
iii) Priority	Shall
iv) Traceability	Shall
B.1.7 Test Case Specification	Shall
a) Test coverage items:	Shall
i) Unique identifier	Shall
ii) Description	Shall
iii) Priority	Shall
iv) Traceability	Shall
b) Test cases:	Shall
i) Unique identifier	Shall
ii) Objective	Should
iii) Priority	Shall
iv) Traceability	Shall
v) Preconditions	Shall
vi) Inputs	Shall
vii) Expected results	Shall
viii) Actual results and test result	Shall
B.1.8 Test Procedure Specification	Shall
a) Test sets:	Shall
i) Unique identifier	Shall
ii) Objective	Shall

ISO/IE/IEEE 29119-3 Information Items	Normative requirement
iii) Priority	Shall
iv) Contents (Traceability)	Shall
b) Test procedures:	Shall
i) Unique identifier	Shall
ii) Objective	Shall
iii) Priority	Shall
iv) Start up	Shall
v) Test cases to be executed (Traceability)	Shall
vi) Relationship to other procedures	Shall
vii) Stop and wrap up	Shall
B.1.9 Test Data Requirements	Shall
a) Detailed test data requirements:	Shall
i) Unique identifier	Shall
ii) Description	Shall
iii) Responsibility	Shall
iv) Period needed	Shall
v) Resetting needs	Shall
vi) Archiving or disposal	Shall
B.1.10 Test Environment Requirements	Shall
a) Detailed test environment requirements:	Shall
i) Unique identifier	Shall
ii) Description	Shall
iii) Responsibility	Shall
iv) Period needed	Shall
B.1.11 Test Data Readiness Report	Shall
a) Test data status:	Shall
i) Unique identifier	Shall
ii) Description of status	Shall
B.1.12 Test Environment Readiness Report	Shall
a) Test environment status:	Shall
i) Unique identifier	Shall
ii) Description of status	Shall
B.1.13 Test Execution Log	Shall
a) Events:	Shall
i) Unique identifier	Shall
ii) Time	Shall
iii) Description	Shall
iv) Impact	Shall
B.1.14 Incident Report	Shall
a) Incident details:	Shall
i) Timing information	Shall
ii) Originator	Shall

ISO/IE/IEEE 29119-3 Information Items	Normative requirement
iii) Context	Shall
iv) Description of the incident	Shall
v) Originator's assessment of severity	Shall
vi) Originator's assessment of priority	Shall
vii) Risk	Shall
viii) Status of the incident	Shall

Annex C
(informative)

Overview of Examples

C.1 Overview

Annexes D to S contain examples of the application of the templates on both agile and traditional projects, to demonstrate the applicability of this standard to both types of projects. It should be noted that these are examples only, and many variations are possible and likely.

Particularly this is the case in agile projects. The reduced (more agile) information items presented in the agile examples are 'lightweight' versions of the information items. This approach is acceptable because of the lower perceived development risks, whereas the 'heavyweight' versions presented for the other examples are related to the higher needs for assurance in the lifecycle.

Any project can tailor documentation from full (all documents) to a minimal set of test documents, where minimal would be project defined.

NOTE The word “shall” appears in some of the example documentation. These “shall”s are example wordings only and not normative.

The example documentation is based around two example projects:

Agile Corporation is a large publication organization producing magazines and books. The Corporation has an internal IT department, which is responsible for all the IT products that are in use by the organization, in supporting them in their business. Projects are run by a single agile team, so there are no projects performed using traditional development methods. The organization has several years of experience working in this way, and finds that it works really well with their needs for new and enhanced IT systems to support the business.

The project featured in this example is the development of a new web-based subscription system allowing people to become subscribers and allowing existing subscribers to change their personal information and order new or extended subscriptions.

Traditional Ltd is a small company that produces advanced analysis equipment for the farming industry. Some of their products are critical, in the sense that wrong analysis results could cause prescription of wrong doses of fertilizer (either too much or too little). The organization is hence required to produce the product according to a specific standard that state requirements concerning production and quality assurance of certain documents and traceability between work product elements.

The project featured in this example is the development of the PC-part of a product called UV/TIT-14 33a. It is an apparatus to measure fertilizer components and their concentration in earth samples. The apparatus has a user interface working on a PC with wireless connection to the measuring system.

Not all of the example documents include the sections for Document Specific Information or Introduction; this is because this information is company-specific and the examples focus on the testing contents of the documents.

The examples might not be internally consistent; each section is to be regarded as an independent example of the information related to the topic (heading).

The examples are not necessarily complete. Where paragraphs have been left out this is marked by three vertical dots, like this:

```
-
-
-
```

Omitted text is identified with an ellipsis, like this "**…**".

Annex D
(informative)

Test Policy

D.1 Example 1 – Agile Corporation

Agile Corporation is a large publication organization producing magazines and books. See more details in the introduction in Annex C.

Test Policy for Agile Corporation, V1.2 (02/13/2009)

Issued by: Ursula Mayers, Head of Development

Approved by: Stephan Blacksmith, Head of QA

Scope: This Test Policy describes the corporate view of testing for Agile Corporation and provides a framework within which all testing carried out on all internal projects within the organization will be performed.

Introduction: Agile Corp recognizes the need for testing of its internal products. The cost of developing high quality software systems can be considered as falling into four categories: prevention costs, testing costs, internal failure costs and external failure costs. It is generally cheaper to prevent defects than detect them and fix them (testing costs plus internal failure costs), and the highest cost is of external failure when detected by the users. To avoid this Agile Corp practices Test-Driven Development (TDD) and Acceptance Test-Driven development (ATDD), which are software development techniques. In its implementation of TDD, Agile Corp uses white-box techniques as defined in ISO/IEC/IEEE 29119 part 4.

Objectives of testing: The objective of testing is to provide sufficient information to determine the current quality of the system under test. As such all activities aimed at achieving this are considered to be software testing activities (e.g. integration, system, acceptance and regression testing).

Test process: The software testing will be based on test processes as defined in ISO/IEC/IEEE 29119-2 and aligned with the development approach.

Test organization structure: Testing will be resourced within the Agile Corp from a central pool of testers who are assigned to the project. In addition, a central 'expert' software testing resource led by the Head of Testing will provide test consultancy services to projects as necessary. Test organization structure inside of a project will follow project guidance.

Tester training: All members of testing teams are expected to have appropriate university education or at least a minimum level of industry certification in software testing. Additionally, testers are expected to be knowledgeable in agile concepts, or to become so within three months of joining a test team.

Standards: Test documentation will be based on ISO/IEC/IEEE standard 29119-3 "Test Documentation", adapted for use in agile projects.

Other relevant policies: Software Development Policy for Agile Corporation, V4.3 (12/12/2008)

Test process improvement and value determination: End of iteration retrospectives will capture lessons learned, metrics, and improvement concepts which will be provided to the central test organization.

D.2 Example 2 – Traditional Ltd

Traditional Ltd is a small company that produces advanced analysis equipment for the farming industry. See more details in the introduction in Annex C.

This policy is published on the Traditional intranet under Management >> Policies. Hence it does not contain all the document related information, and it is not versioned, but a publishing date is visible.

Test Policy

Objective and definition of test

At Traditional Ltd, testing is considered as a means to achieving user and customer confidence in our products. Testing is one of many means to achieve this goal.

Testing process

Any software project must include a test project. In other words, the test project must be a subproject of a corresponding software project.

The two projects should be started at the same time. The test process includes the activities: planning, analysis and design of test material, execution and recording of the test including registration of any incidents, and test completion and reporting. Testing is affecting something (object under test), observing the effect and deciding whether this effect is considered correct or incorrect behaviour.

Organization

Each project will be staffed by analysts, designers, programmers, and test analysts. They will all report to the project manager. Students could be hired to execute tests.

Evaluation of testing

For each product, management should decide which level of quality is to be achieved expressed as the maximum number of incident reports over a given period from customers.

At the release of a product the test group must deliver a report on the product’s expected behaviour. One year after release, management reviews this report and compares it with respect to feedback from the market (number of incident reports, number of failures.)

Standards

We follow our own standards found on the intranet. These are all based on ISO/IEC/IEEE standard 29119-3 "Test Documentation".

Policies in Traditional Ltd

The policies for Software Development and Quality Assurance form the basis for all software development and testing in Traditional Ltd.

Approach to test process improvement

At the release of a product the test group must deliver a report which analyses the project from a testing point of view. Any improvements suggested in this report are discussed with management in order to decide which improvements to make.

When the evaluation of the testing takes place one year after release, management considers if improvements should take place.

Annex E
(informative)

Organizational Test Strategy

E.1 Example 1 – Agile Corporation

Agile Corporation is a large publication organisation producing magazines and books. See more details in the introduction in Annex C.

Organizational Test Strategy for Agile Corporation, V1.1 (03/23/2009)

Issued by: Ursula Mayers, Head of Development

Approved by: Stephan Blacksmith, Head of QA

Issuing organization: The head of testing at Agile Corporation is responsible for preparing the Organizational Test Strategy. Upon review and approval, the senior management at Agile Corporation is responsible for distributing the Organizational Test Strategy.

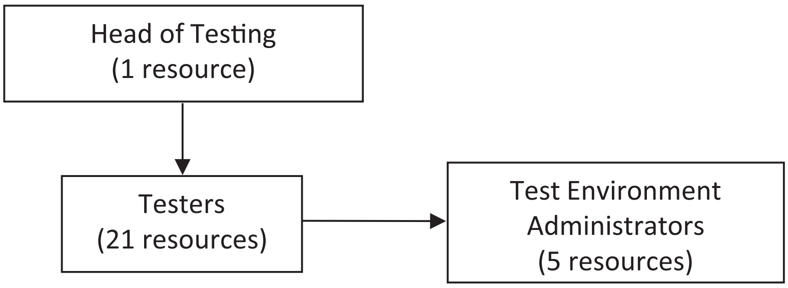
Scope: This Organizational Test Strategy provides the Corporation's overall approach to testing. We have developed and implemented several guidelines that are applicable across all projects. We aim to provide testing at every point in the system and software lifecycle. This is accomplished by having our test group engaged early in the lifecycle process, participating in teams with developers, and working with user stories even in a draft state. These pertinent artefacts provide the basis for establishing test plans and scoping the test effort. In addition to the development of test plans, the organization will use agile test activities such as stakeholder involvement in test design, preparing test automation, peer reviews, various testing design techniques (project applicable), lightweight defect tracking and reporting.

References: Agile Manifesto

Generic risk management: All risk management must follow the prescribed corporate Risk Management process as defined in Corporate Policy-RM56, where the general risk register is identified. Any deviations and waivers must be approved by senior management.

Degree of independence: The Corporation test organization is led by the head of testing who has no direct association with the head of development. The test organization is technically, managerially, and financially independent from the development organization of Agile Corp, while within a project assigned testers could participate directly in self-organized teams, which include development.

Test organization structure: Agile Corporation Test Organization has a pool of independent test professionals from which testers are assigned to agile teams, e.g. a scrum team, where testers are members of the overall team.



Test documentation strategy: The Test Organization will comply with the test documentation defined in ISO/IEC/IEEE 29119-3, and the principles of agile development. Any deviations will require the approval of the head of testing.

Test sub-processes documented in project test plans: The Corporation relies on our highly competent head of testing to help ensure the most effective type of testing is invoked. This is accomplished via a mentoring program with the testers on scrum teams, and it includes functional and non-functional methods, test design techniques, and testing tools, tailored from ISO/IEC/IEEE 29119-1, -2, and -4. Additionally each project defines test selection, priority and management. Further, the project must select their own test environments, retest/regression practices, and incident management practices. These items are agreed during continuous direct interaction with stakeholders over the life of each project. The level of test plan documentation (size and format) is also agreed with project stakeholders.

E.2 Example 2 – Traditional Ltd

Traditional Ltd is a small company that produces advanced analysis equipment to the farming industry. See more details in the introduction in Annex C.

Traditional Ltd has an organizational test strategy with a project-wide part, and a part for each test sub-process. This example only includes the project-wide part and the parts for component test and system test.

Organizational Test Strategy

Organizational Test Strategy	
Issue	**Strategy**
Generic risk management	The risk management in a project must be based on the generic risk register for the specific type of project and the generic risk management process. The risk registers are found in XX. When a project is closing down the relevant generic risk register(s) must be updated as appropriate.
Test selection and prioritization	Test cases and test procedures will be prioritized according to the risk associated with the requirements the cases are covering. If a test procedure includes test cases with different risk levels, the test case with the highest level determines the risk level for the entire procedure. Execution of test procedures must always be scheduled according to the risk, so that the higher the risk level the sooner the procedure is scheduled to be executed. Care must however be taken so that all feature sets are covered by some testing, that is no feature set must be left out of the execution schedule.
Test documentation and reporting	The test projects must be documented in such a way that an audit can establish what has been planned and what has been performed. Tracing between artefacts is essential. A project test plan and a project test completion report as outlined in ISO/IEC/IEEE 29119 Part 3 must be produced at the test project level.
Test automation and tools	The test management tool BCG is to be used on all test projects and for all sub-processes. In the cases where more than 4 regression tests are planned the project might consider using a capture/playback testing tool.
Configuration management of test work products	The Traditional Ltd process for configuration management must be followed for all test work products.
Incident management	The Traditional Ltd process for incident management must be followed.
Test sub-processes	Each test project must include the following test sub-processes: - Performance test – if applicable in relation to requirements - Operability test - Component testing - Component integration testing – preferably bottom-up - System testing

Organizational System Testing Strategy	
Issue	**Strategy**
Entry and Exit criteria	The integration test completion report and the system test specification must be approved before the system test execution might begin. All the system test deliverables must be approved before the system test is finished.
Test completion criteria	The system test is supposed to achieve 100 % requirements coverage, and all test procedures must be executed without incidents.
Test documentation	A system test plan and a system test completion report as outlined in ISO/IEC/IEEE 29119 Part 3 must be produced, as must all documents defined for dynamic testing.
Degree of independence	The system test must be specified by the staff in the test department and executed by students.
Test design techniques	Appropriate black-box test case design techniques are to be used. Error guessing could also be used, if defect information exists for previous versions.
Test environment	The system testing environment must be identical to the production environment in terms of hardware and software. In the case of embedded systems the system test could be executed on a simulator. Data could be made anonymous, but must otherwise be 100 % representative.
Metrics to be collected	The following shall be reported in the system test completion report: - Total number of specified test procedures - Total number of executed test procedures - Total number of testing hours spent on specification - Total number of hours spent on execution and registration of incidents - Total number of hours elapsed for testing - Total number of failures found
Retesting and regression testing	All test procedures resulting in incident reports must be rerun after defect correction. Regression testing during the system test sub-process is at the test manager's discretion. In the final system test run, all test procedures must be executed.

Organizational Component Testing Strategy	
Issue	**Strategy**
Entry and Exit criteria	The test item (component) must compile and link, and the component test specification must be approved before the component test execution can begin. All the component test deliverables must be approved before the component test is finished.
Test completion criteria	The tests for each component is supposed to achieve at least 90 % statement coverage and at least 80% decision outcome coverage, and all test cases for a component must be executed without incidents. Reason for non-conformances must be reported and accepted by the project manager.
Test documentation	A component test plan and a component test completion report as outlined in ISO/IEC/IEEE 29119 Part 3 must be produced; Test cases to help ensure coverage must be produced for each component.
Degree of independence	The component test must be specified and executed as a peer test, that is, by a developer who is not the one who coded the component under test.
Test design techniques	Appropriate black box test case design techniques are to be used, and these must be supplemented by the white box techniques: statement testing, and decision outcome testing, where necessary to help ensure required coverage.
Test environment	The component testing can be executed in the development environment of the developer who is designing the test, i.e. not the environment of the developer who coded the component under test.
Metrics to be collected	The following must be reported in the component test completion report: • Average obtained statement coverage. • Average obtained decision outcome coverage. • Total number of incidents found and corrected.
Retesting and regression testing	Each component must be retested until completion criteria have been reached.

Annex F
(informative)

Test Plan

F.1 Example 1 – Agile Corporation

Agile Corporation is a large publication organization producing magazines and books. See more details in the introduction in Annex C.

This plan is available on the project portal and the newest version is also posted in the top right corner of the story board in the development room.

Test plan for: New subscription system (NSS) **Vers.:** Iteration 3

Covers: NSS iteration 3 result and stories, including result of previous iterations.

People: Each iteration is carried out by the team consisting of developers, user representatives, and testers. The developers ultimately refer to the Head of Development (Ursula), and the testers to the Head of QA (Stephan).

Risks: The specific risks for this iteration are listed on the story cards. The general risk is that the iteration team does not have access to live data in the supporting databases.

Test strategy: Remember to:

- Create automated tests based on stories before coding starts, test the new code, and test the integration with the current version of the system before marking a story as completed.
- Retest every time something has been changed in the result from previous iterations as well as for this current iteration, and to regression test the entire result of this iteration before the showcase meeting.
- Estimate and cost testing and development to fit within the allocated iteration as agreed at the iteration kick-off, and return to the backlog any items which cannot be met by the end, including any technical debt (errors) accumulated which cannot be resolved in the allocated iteration.
- Use the test design technique(s) most appropriate to the acceptance criteria, keeping in mind that higher risk stories require more thorough testing that lower risk stories.
- Ensure and verify that testing achieves statement coverage of at least 90%, of all code as well as branch coverage for 80% of the high risk stories and 60% of the low risk stories.
- Ensure that no defects of severity 1 or 2 remain outstanding in the implementation of a story before it is integrated.
- Define customer facing (acceptance) ATDD Test in the iteration with customer/user agreement and participation.
- Before the showcase meeting, test the result of the iteration in the official test and presentation environment.

To be continued

Continued from previous page

- Cover test items at daily stand up meetings, including low level test plan activities and risk documented on whiteboards.
- Store all test scripts in the ABC tool, so that they are available for retesting and regression testing as needed.
- Issue a short summary report of the testing at the end of each iteration, and place it in the project portal.

Testing activities and estimates: Test effort is expected to take one third of the total team effort spent during the iteration. At this point in time the showcase test is estimated to take 3 hours duration.

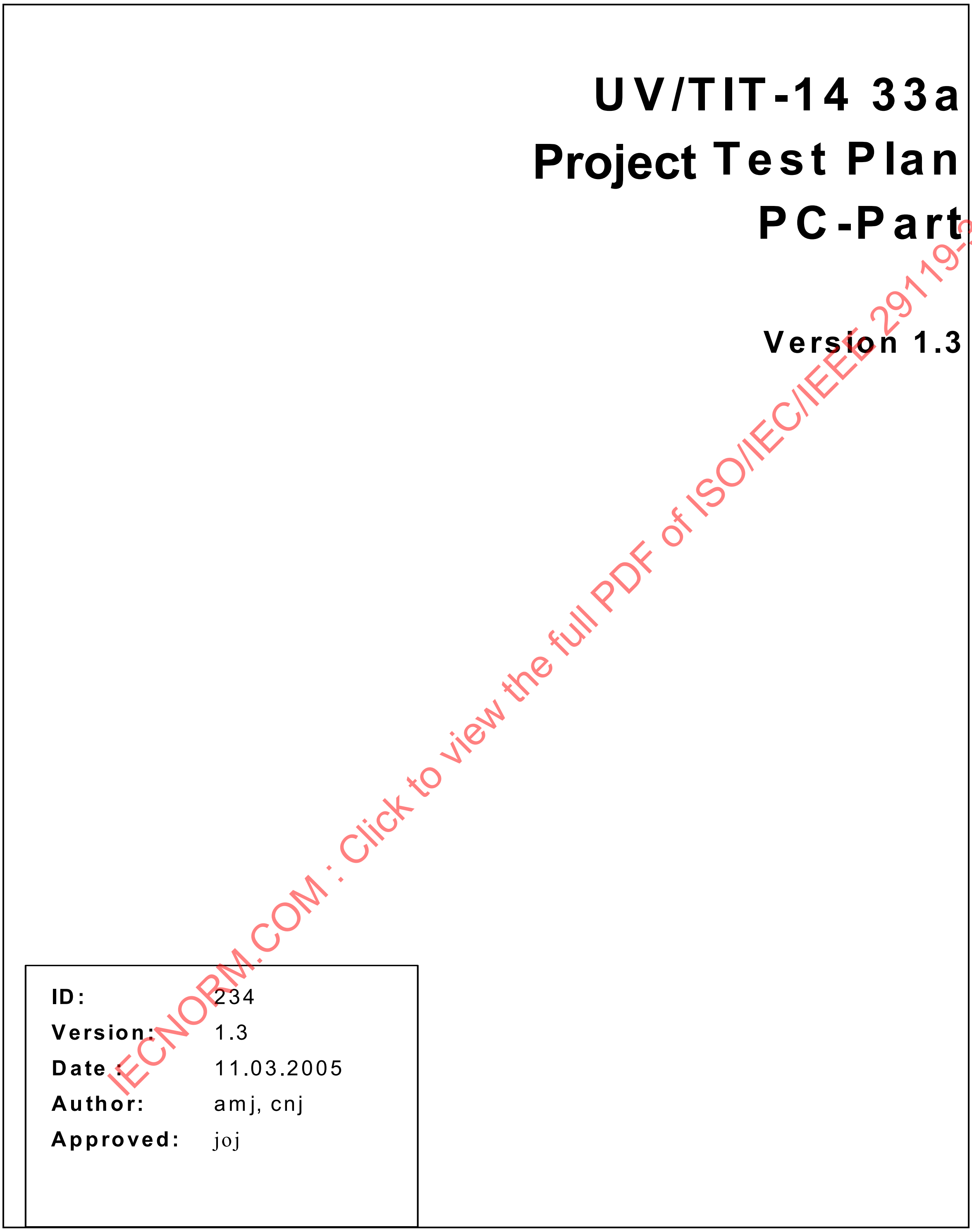
F.2 Example 2 – Traditional Ltd

This example includes two sub-examples of test plans, namely:

- Project Test Plan
- System Test Plan

F.2.1 Project Test Plan

Traditional Ltd is a small company that produces advanced analysis equipment to the farming industry. See more details in the introduction in Annex C.



UV/TIT-14 33a
Project Test Plan
PC-Part

Version 1.3

ID:	234
Version:	1.3
Date :	11.03.2005
Author:	amj, cnj
Approved:	joj

Revision log

Date	Version	Changes	Init.
10.10.03	0.1	Initial issue of document	amj
12.10.03	0.2	New test item description	cnj
16.01.04	1.0	Correct typos after review	cnj
20.01.04	1.1	New risk identification	amj
05.02.04	1.2	Minor change in structure of text	cnj
11.03.05	1.3	Update after walkthrough	amj

1 Introduction

1.1 Scope

The purpose of this document is to provide the information and the framework required to plan and perform all test processes needed for the testing of the PC-part of the UV/TIT-14 33a product.

1.2 References

[PP] Project Plan

[PRS] Project Requirements Specification

[OTS] Organizational Test Strategy for Traditional Ltd

[KD] Requirements Specification for PC part of UV/TIT-14 33a.; Vers. 1.8

[HW/SW-spec] Hardware and software specification

1.3 Glossary

The definitions specified in [PP] are also valid in this document.

The following abbreviations are used:

TBD	To be defined

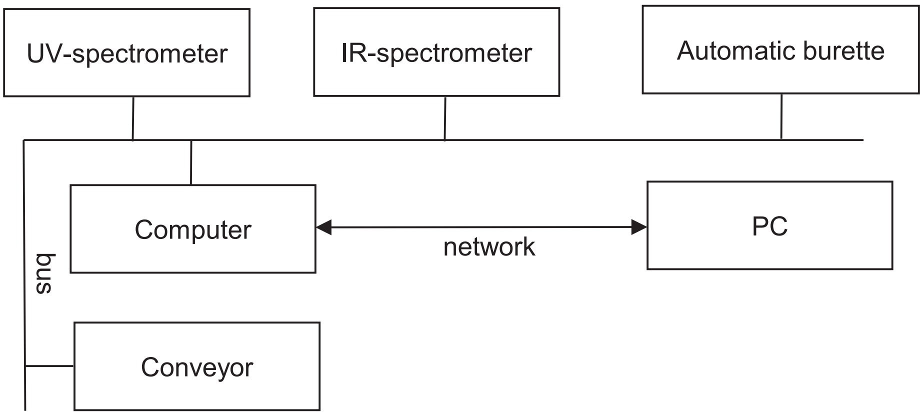
2 Plan context

2.1 Project

The UV/TIT product consists of the following hardware modules:

- UV spectrometer;
- IR Spectrometer;
- Automatic burette;
- Conveyor;
- Computer (server);
- PC.

The architecture is shown in the following figure.



The system consists of the following software modules on the computer (the server):

- UV module;
- IR module;
- Burette module;
- Conveyor module;
- Network module.

The system consists of the following software modules on the PC:

- Calibration module;
- Compound identification and concentration module;
- Setup module;
- Control and report module;
- Network module.

2.2 Test item(s)

The test for this project includes testing of:

- Each module of the PC software, as listed in 2.1 above;
- Each component of the PC software modules listed in 2.1 above;
- The functionality of the complete software system.

The exact versions of the various test items must be obtained from the configuration management system at the time of specifying the test, and must be controlled prior to the execution of any test.

2.3 Test scope

The PC system consists of software modules listed above. The network module is bought as a standard product and tested by so many organizations that it is not considered worthwhile testing. All the other modules shall be tested under the assumption that the operating system on the PC and the network are working correctly.

Functionality directly related to the network connection to the computer will not be tested, except indirectly when these features are used in connection with some other tests.

Non-functional quality factors like performance, security, safety, and usability will not be tested in this test project because these tests will be outsourced to another company that will perform this part of the test. A separate Test Plan for these will be created by the outsourcing company responsible for that testing.

2.4 Assumptions and constraints

None.

2.5 Stakeholders

Please refer to the stakeholder analysis in [PP].

3 Testing communication

Please refer to [PP].

4 Risk register

The following abbreviations are used in the risk tables:
P = probability or likelihood of the risk
I = Impact or effect if risk materialises
E = Exposure = Probability x Impact

The scales for both probability and impact will be 1-6, where 6 is the highest.

4.1 Product risks

Risk ID	P	I	E	Mitigation Activities
1 The calibration is not correct.	2	5	10	Review of design and of code Extra thorough component test Investigate the possibility of automated test recording (checking actual results against expected – maybe get an ‘oracle’)
2 The compound identification is not correct.	2	6	12	Review of design Code inspection Extra thorough component test
3 The concentration calculation is not correct.	1	6	6	Review of design Code inspection Extra thorough component test
4 The calculation ‘drifts’ if the machine is not turned off now and again.	3	5	15	Code inspection Dynamic analysis to identify any memory leaks Stress test by letting the machine operate for several days without turning it off
5 The compound identification is too slow.	3	1	3	Performance test under various conditions
6 The reports are not correct.	2	5	10	Review of design and of code Extra thorough component test Investigate the possibility of automated test recording (checking actual results against expected – maybe get an ‘oracle’)
7 A software crashes in the middle of an analysis will give unreliable results.	2	2	4	Stress test by forcing the machine to crash during an analysis
8 The user manuals are not understandable for laboratory technicians.	4	3	12	Usability assessment of user manuals Inspection of user manuals

Risk ID	P	I	E	Mitigation Activities
9 The setup is difficult to understand.	4	2	8	Usability assessment of prototype of interface Extra care in requirements specification Review of requirements Review of design Extra careful component test
10 There are problems with the text in localised reports.	5	3	15	Review of all reports in languages known to create longer texts

4.2 Project risks

Risk ID	P	I	E	Mitigation Activities
1 Insufficient staff available	2	4	8	Be extremely careful when estimating the effort of the project. Carry out a thorough product risk analysis and plan in strict accordance with it. Follow test progress extremely closely. Report progress and resource problems clearly at every available opportunity.
2 The people who are available lack sufficient knowledge and experience	2	3	6	Perform a gap analysis of needs in relation to what is available. Prepare a training plan for each individual participant. Include training time in the schedule. Find mentors, if possible. Set aside extra time for review of work carried out by people with little or no experience.
3 Some of the people who are available will not / cannot work together	5	2	10	Try to identify what the problem is. Arrange for arbitration if necessary / practicable. Where necessary, arrange for a Belbin analysis to increase understanding between the various types. Distribute activities so that there is as little contact between the individuals in question as possible.
4 There are too few licenses available for test execution tools	3	2	6	Attempt to convince those responsible for approving additional licenses that this is necessary. Distribute and plan activities in detail in order to reduce waiting times as far as possible. Perform meticulous progress follow ups. Report related problems at an early stage and as clearly as possible.
5 The test manager is not familiar with the tools used to support management of the test project	4	1	4	Obtain funding for and plan a course if possible. Find someone from within the company who has experience of using the tool. Allocate additional time to the estimation of test management activities. Report related problems at as early a stage as possible and as clearly as possible.

Risk ID	P	I	E	Mitigation Activities
6 Previous experience shows that the sub-contractor doesn't always supply the expected material on time and to the expected quality	3	4	12	State relevant contractual requirements precisely. Establish specific quality criteria for deliverables. Specify the consequences of breaches of quality and scheduled delivery. Follow progress and quality closely. Implement consequences as/when necessary.
7 We are using more than 30% of test execution time on incident reporting	4	5	20	Suspend testing until unit test has been done.

5 Test strategy

5.1 Test sub-processes

The test for the PC-part of the UV/TIT-14 33a product shall include the following test sub-processes:

- Component testing;
- Component integration testing;
- System Testing.

5.2 Test deliverables

For each test sub-processes the following documentation must be produced:

- Test sub-process test plan;
- Test specification;
- Test log;
- Test sub-process completion report.

5.3 Test design techniques

This is specified for each test sub-process according to the appropriate part of the [OTS].

5.4 Test completion criteria

This is specified for each test sub-process according to the appropriate part of the [OTS].

5.5 Metrics to be collected

This is specified for each test sub-process according to the appropriate part of the [OTS].

5.6 Test data and test environment requirements

This is specified for each test sub-process according to the appropriate part of the [OTS].

Specific testing tool requirements are:

- Ant;
- JIRA;
- JBoss;
- Test link 1.8 RC2.

5.7 Retesting and regression testing

This is specified for each test sub-process according to the appropriate part of the [OTS].

5.8 Suspension and resumption criteria

Suspension criteria are listed in the project risk register above (see Risk 7).

5.9 Deviations from the organizational test strategy

See individual test sub-process test plans.

6 Testing activities and estimates

This information is available in the intranet tool Mpower and Organization Dashboard measures.

https://mpower.Traditional.com/irj/portal

For information on cost related measures and monthly tracking, refer the following link:

https://processnet.masked.com/projectdashboard/Dashboardhome_new.asp

This information is only accessible by project managers and above.

7 Staffing

7.1 Roles, activities, and responsibilities

The high-level activities are the sub-processes to be performed. The detailed activities and responsibilities will be documented in the test sub-process test plans.

7.2 Hiring needs

See individual test sub-process test plans.

7.3 Training needs

See individual test sub-process test plans.

8 Schedule

The schedule for the testing covered by this plan is included in the Gantt diagram for the project.

F.2.2 System Test Plan

UV/TIT–14 33a PC-Part, System Test Plan

The front page and document specific information is not included in this example.

1 Introduction

See the Project Test Plan

2 Plan context

2.1 Project

See the Project Test Plan.

2.2 Test item(s)

The test item is the integrated PC-software for the PC-part of UV/TIT-14.

2.3 Test scope

The features to be tested can be subdivided into these groups.

- Setup of the system;
- Identification of compounds (IR+UV);
- Concentration of compounds (UV + control of burettes);
- Calibration of UV, IR, and burettes;
- Reports on identification, concentrations, and calibration;
- Reports on setup;
- Control of conveyor system (speed, correct start and stop positions etc.);
- Statistics.

This test plan does not cover other test sub-processes than system testing, i.e. not for example component testing and acceptance testing.

Features directly related to the network connection to the computer will not be tested, except indirectly when these features are used in connection with some other tests.

Non-functional quality factors like performance, security, safety, and usability will not be tested in this test project because these tests will be outsourced to another company that will perform this part of the test.

The test covers all the PC-software developed specifically for this system, which means that other elements like the operating system and the network are not tested explicitly.

2.4 Assumptions and constraints

See the Project Test Plan.

2.5 Stakeholders

See the Project Test Plan.

3 Testing communication

See the Project Test Plan.

4 Risk register

4.1 Product Risks

See the Project Test Plan for product risks.

4.2 Project Risks

D = Development

T = Test

Risk ID	Consequences	Prevention	Mitigation
1 Dependency of the development.	If the development is delayed the test will be delayed accordingly. It might be difficult to keep to the schedule.	D: Perform a realistic re-estimation and planning. T: None	D: Re-evaluate plans to sufficient extent, not "slice by slice". T: None

Risk ID	Consequences	Prevention	Mitigation
2 It appears that we can't automate as many test procedures as anticipated.	More test procedures must be executed manually. It might be difficult to keep the schedule.	D: None T: Examine if this is the case during developing of test procedures.	D: None T: Add resources.
3 Presence of blocking faults	Postpone some parts of the test. It might be difficult to keep the schedule.	D: Take care of thorough module and integration test T: None	D: Correct faults. T: None
4 Changes keep coming all the time.	The test effort is used to update documentation rather than testing.	D: Baseline the specifications and respect this baseline. T: None	D: None T: None

5 Test strategy

5.1 Test deliverables

The test deliverables for the entire system test are:

- this present plan in an up-to-date version at the time of delivery;
- a full set of test specifications;
- a test completion report for the complete system test of the PC-Part of UV/TIT-14 33a.

The test deliverables for each executed test procedure are:

- a test log signed by the test manager. The test log must include the identification numbers of the incident reports raised during the execution of the test, if any;
- an updated version of the test specification or a list of known defects in the test specification.

5.2 Test design techniques

The following test case design techniques are to be used where they are relevant:

- equivalence partitioning and boundary value analysis;
- classification tree method;
- decision table testing;
- state transition testing;
- use case testing.

5.3 Test completion criteria

The system test must achieve 80 % requirements coverage, and all test procedures must be executed without failures of severity 1 (High).

5.4 Metrics to be collected

The following metrics are to be collected during the course of the system test:

- number of executed test cases;
- number of incidents per category;
- number of re-executed test cases;

- number of solved incidents per category;
- number of hours spent.

5.5 Test environment requirements

The tester (the person responsible for the test execution) must have the following documents available during the execution of the test:

- [PRS]
- this test plan
- user manual for UV/TIT-14 33a
- copies of the test procedures for each of the tests to be executed, for use as test log sheets

The hardware and the operating system and the network are all described in [HW/SW-spec]

In order to conduct the tests another PC will be used as a simulator, so that the commands sent to the computer can be analyzed and simple responses can be simulated. This will reduce manual handling of many samples, but it requires the development of a simulator. So there is a need for a simulator, which runs on an ordinary PC and is identical to the system PC.

5.6 Retesting and regression testing

The necessary retesting and regression testing to fulfil the completion criteria must be performed. It is estimated that at least 3 testing cycles will be performed, the last one including a complete regression test.

5.7 Suspension and resumption criteria

If test completion is impossible due to external causes the completion must be postponed until these have been cleared. It must be evident from the test log what has happened and for how long the testing was suspended. As little as possible of the testing already done should be repeated at resumption, based on a risk assessment.

If the completion of a test suite is impossible due to a failure, this must be reported via the incident management system, and the failure must be assigned a test severity of ‘High’. At resumption of the test the affected test procedure must be repeated.

5.8 Deviations from the organizational test strategy

The Organizational Test Strategy requires 100 % requirements coverage, but this has been reduced to 80 % for this system test because there are relatively few product risks and the component test is planned to be very thorough.

6 Testing activities and estimates

The test work will be broken down into the following main activities in accordance with [OTS]:

1. Definition of an overall structure for the test in the form of feature sets to be tested;
2. Detailed specification of the test cases and test procedures;
3. Establishment of the test environment;
4. First execution cycle of test procedures;
5. Second cycle of test procedures (retest and regression test from first cycle);
6. Third cycle of test procedures (retest and regression test from second cycle and any left over from first cycle);
7. Status report about the test execution every week;
8. Test completion reporting.

The detailed testing activities and their estimates can be found in SYS-TEST.xls in the project portal.

7 Staffing

7.1 Roles, activities, and responsibilities

The RACI matrix below illustrates which role is involved in which activity(ies) and what the involvement is. The activity numbers refer to the activity list above

Role \ Activity	1	2	3	4
Test lead	A	A	A	A
Test analyst	R	R	R	C
Test designer	R	R	R	I
Test environment expert	-	R	-	I
Test executor	I	I	I	R

Responsible **A**ccountable **C**onsulted **I**nformed

Resource	Name	Description
LN	Leila Nielsen	Test analyst
CBB	Christina Bagge	Test analyst / designer
CD	Carsten Dominique	Test environment expert
T1 – T2	TBD	Test execution

The roles are filled as follows:

7.2 Hiring needs

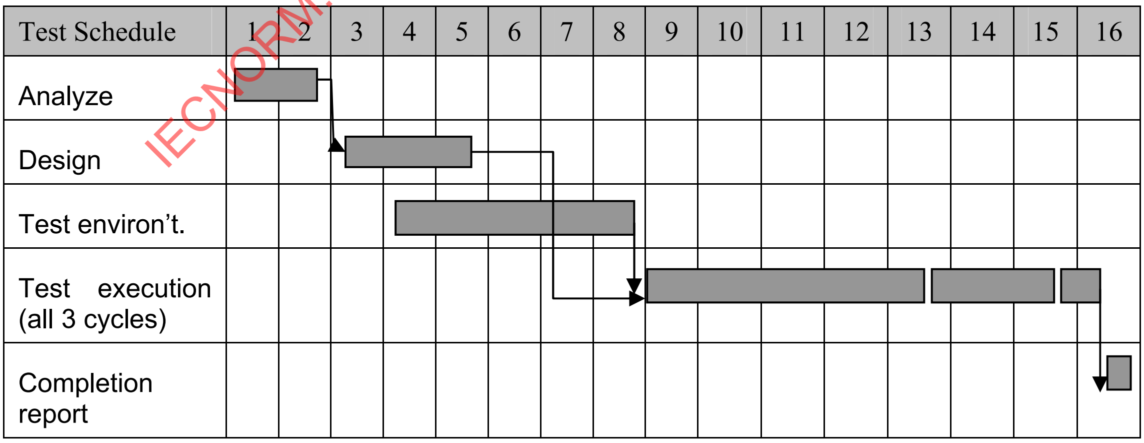
We need two students (or similar) to execute the tests. These will be hired in accordance with the HR hiring rules.

7.3 Training needs

Only an introduction to the system for the 2 test executors is needed. This is estimated to take 1 hour on their first day.

8 Schedule

The overall schedule for the test is shown below.



Annex G
(informative)

Test Status Report

G.1 Example 1 – Agile Corporation

A status report is produced at the end of each iteration in the form of a summary report placed in the project portal.

Summary Status Test Report for: New subscription system (NSS) **Vers.:** Iteration 3

Covers: Complete NSS iteration 3 results.

Progress against Test Plan: Test has been done in the iteration on the 5 user stories for this iteration.

For the one high risk story 92 % statement coverage was achieved, and for the others 68 % statement coverage was achieved on average.

There are no outstanding defects of severity 1 and 2, but the showcase showed that the product has 16 defects of severity 3.

Factors blocking progress: None

Test measures: 6 new test procedures have been developed, and 2 of the other test procedures have been changed.

The testing in the iteration has taken up approx. 30 % of the time. The test took about 2½ hours.

New and changed risks: The risks for the stories have been mitigated satisfactorily. New risks are not identified yet.

Planned testing: As per test plan.

Backlog added: 16 defects (severity 3)

G.2 Example 2 – Traditional Ltd

Traditional Ltd is a small company that produces advanced analysis equipment to the farming industry. See more details in the introduction in Annex C.

Project PC-part of the UV/TIT-14 33a product.

System Test Status Report, V 1.0, 22.03.2004

Test Status on 22nd March 2004

Reporting period: 15 – 21 Mar 2004

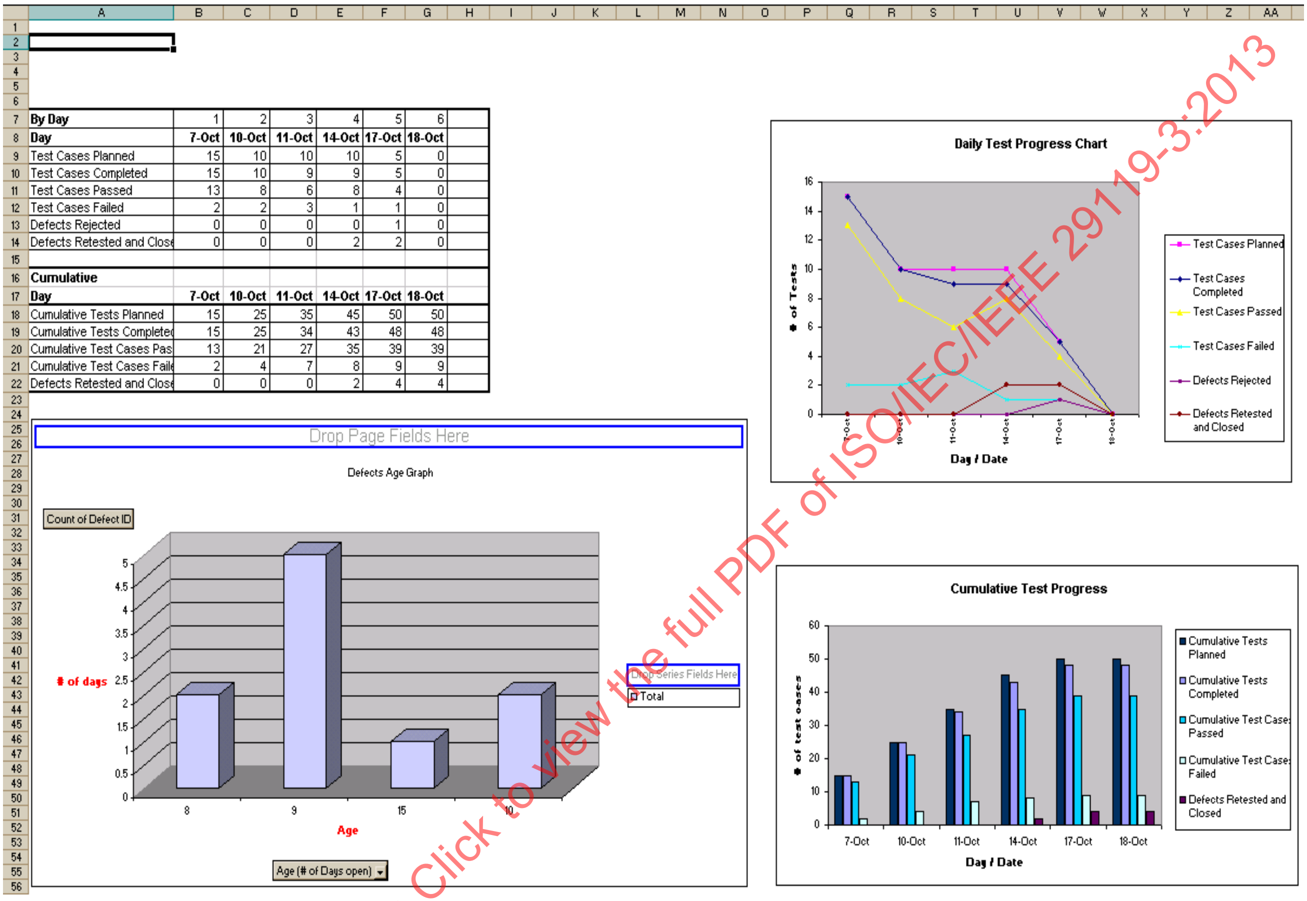
Progress against test plan: The functionalities of the ‘XX’ module of the application were tested.

We almost reached the target in the plan of having executed 2/3 of the test cases. We expect to catch up with the small delay next week. It appears that most of the failures founds are due to simple defects in individual modules. See details below.

Blocking factors: None

Test measures:

NOTE The graphs do not necessarily correspond to the tables in this example; they are included to illustrate that graphs and tables could be included in a report.



By Day	1	2	3	4	5	6
Day	**7-Oct**	**10-Oct**	**11-Oct**	**14-Oct**	**17-Oct**	**18-Oct**
Test Cases Planned	15	10	10	10	5	0
Test Cases Completed	15	10	9	9	5	0
Test Cases Passed	13	8	6	8	4	0
Test Cases Failed	2	2	3	1	1	0
Defects Rejected	0	0	0	0	1	0
Defects Retested and Close	0	0	0	2	2	0
Cumulative						
Day	**7-Oct**	**10-Oct**	**11-Oct**	**14-Oct**	**17-Oct**	**18-Oct**
Cumulative Tests Planned	15	25	35	45	50	50
Cumulative Tests Completed	15	25	34	43	48	48
Cumulative Test Cases Pas	13	21	27	35	39	39
Cumulative Test Cases Faile	2	4	7	8	9	9
Defects Retested and Close	0	0	0	2	4	4

The measures collected in the following tables were collected. P1-P6 are priorities, where P1 is highest.

Test Case Execution Measures – Daily

Description	#	%
Total Number of Test Cases Executed today - Actual	**30**	
Number of Test Cases Passed	**27**	**90%**
Number of Test Cases Failed	**3**	**10%**

Test Case Execution Measures – Cumulative

Description	#	%
Total Number of Test Cases Planned	151	100%
Total Number of Test Cases Executed To Date	38	25.17%
Total Number of Test Cases Passed out of executed ones	34	89.47%
Total Number of Test Cases Failed out of executed ones	4	10.53%
Total Number of Test Cases Pending	113	74.83%

Defect Summary - Daily

Description	Total Number	P 1	P 2	P 3	P 4	P 5	P 6
Total # of Defects For the Day	5	0	0	4	1	0	0
# of Defects New	5	0	0	4	1	0	0
# of Defects Retested & Closed	8	0	0	4	3	1	0
# of Defects Re-opened	1	0	0	0	1	0	0

Defect Summary - Cumulative

Description	Total Number	P 1	P 2	P 3	P 4	P 5	P 6
# of Defects To Date	42	0	9	21	10	2	0
# of Defects Rejected To Date	1	0	0	1	0	0	0
# of Defects Fixed To Date	31	0	7	16	6	2	0
# of Defects retested & Closed To Date	32	0	7	16	7	2	0
# of Defects Open\Pending on Date	10	0	2	6	2	0	0

New and changed risks: None

Planned testing: The team is testing and investing a lot of time in uncovering defects that were supposed to have been found during the component testing sub-process. If adequate steps could be taken by the development team in fine tuning the component test cases then the QA team would be able to save a lot of time and concentrate on other aspects of efficiently testing the application. However, the overall Test summary suggests that there are no P1 bugs. If the open P2 could be fixed by the next build then the QA team could still have time for a complete cycle of regression testing and would be able to meet the release time line.

The cumulative test report as shown above suggests that the application did grow stable towards the 2nd cycle of testing. After the completion of the 2nd cycle it can be predicted that all the test cases of 'XX' module would pass.

Annex H
(informative)

Test Completion Report

H.1 Example 1 – Agile Corporation

Agile Corporation is a large publication organization producing magazines and books. See more details in the introduction in Annex C.

This report is available on the project portal and the newest version is also posted in the bottom right corner of the story board in the development room as it is developed/updated.

Test report for: New subscription system (NSS) **Vers.:** Iteration 3361

Covers: NSS final iteration result, including result of previous iterations, in preparation for a major customer delivery (for use).

Risks: The live data risk was retired by creation of a simulated database using historic live data “cleaned” by the test team and customer.

Test Results: Customer accepted this release of the product based on:

16 user stories were successful, including one added after the last status report.

100% statement coverage was achieved in technology-facing testing with the one high risk story, and for the others 72% statement coverage was achieved on average.

Team accepted the backlog on 4 defects of severity 3.

Showcase was accepted by the customer with no added findings. Showcase demo iteration features interfaced with “live” data.

Performance of the iteration features was found to be acceptable by team and customer.

New, changed, and residual risks: Security of the system could become an issue in future releases, assuming a follow work activity is received from the customer.

Notes for future work from retrospective:

Iteration team feels a new member could be needed given possible new risk since no one has knowledge in this area.

Severity 3 defects that move on to backlog should be addressed in next release to reduce technical debt.

The modified live data worked well and should be maintained.

Test automation and exploratory testing is working, but additional test design techniques should be considered, e.g. security and combinatorial testing.

H.2 Example 2 – Traditional Ltd

Traditional Ltd is a small company that produces advanced analysis equipment to the farming industry. See more details in the introduction in Annex C.

Project PC-part of the UV/TIT-14 33a product

System Test Completion Report, V 1.0, 08.11.2004

Written by Test Manager Carlo Titlefsen

Approved by Project Manager Benedicte Rytter

Summary of testing performed:

- The test specification was produced; it included 600 test procedures.
- The test environment was established according to the plan.
- Test execution and recording were performed according to the plan.

Deviations from planned testing: The Requirement Specification was updated during the test to V5.6. This entailed rewriting of a few test cases, but this did not have an impact on the schedule.

Test completion evaluation: All test procedures are executed without failures of severity 1 (High). This has not been reached, because one test procedures was not executed. The requirement this test procedure covers is, however, of such low risk exposure that the test has been accepted by the Product Owner.

Factors that blocked progress: None

Test measures:

One (Test procedure 4.7) of the 600 planned test procedures was not executed at all because of lack of time. All the 599 test procedures that were run had passed at the end of the 3 weeks.

During the test 83 incidents were found and 83 were solved. The reported incidents were number 107 to number 189.

Working hours spent:

- 164 working hours were spent on the production of the test specification
- 10 working hours were spent on the establishment of the test environment
- 225 working hours were spent on test execution and recording
- One half hour was spent on this report.

New, changed, and residual risks: All the risks listed in the test plan have been eliminated, except the one with the lowest exposure, Risk no. 19.

Test deliverables: All deliverables specified in the plan have been delivered to the common CM-system according to the procedure.

Reusable test assets: The test specification and the related test data and test environment requirements could be reused for maintenance testing, if and when this is needed.

Lessons learned: The test executors should have been given a general introduction to the system; it sometimes delayed them finding out how to perform a test case, but fortunately the test analyst was available throughout the test period.

Annex I
(informative)

Test Design Specification

I.1 Example 1 – Agile Corporation

Agile Corporation is a large publication organization producing magazines and books. See more details in the introduction in Annex C.

This test design specification is available on the project portal and the newest version is also posted in the top right corner of the story board under the test plan in the development room.

Test design

We have the following themes:

1. Administration
2. New and extended subscriptions
3. General access on the web-site
4. Complaints

For the showcase testing, stories should be sorted by theme. The test conditions on the back of the story cards, which describe the story acceptance criteria, should be covered.

Version: 1 (Ursula)

I.2 Example 2 – Traditional Ltd

Traditional Ltd is a small company that produces advanced analysis equipment to the farming industry. See more details in the introduction in Annex C.

Test Design Specification

To make it easier to understand what the test specification is based on, an extract of the system requirements for the PC part of UV/TIT-14 33a is included here.

System Requirements Specification for PC part of UV/TIT-14 33a (extract)

4.1 Setup

4.1.1 [22] The system shall have a setup menu with the following menu points:

- Conveyor setup
- Calibration setup
- ... setup

4.2 Setup of the conveyor

4.2.1 [34] The following parameters should be set *[the numbers in brackets are the ranges that can be used]*:

- Max speed (mm/s) (5-50)
- Min speed (mm/s) for a running conveyor (it is 0 for a stopped one) (2-10)

4.2.2 [36] When a new setup is committed the system shall show one of the following messages

- Setup done
- Max speed out of range
- Min speed out of range

4.2.3 [37] The system shall allow the user to leave the setup functionality without changing anything

.
.
.

4.8 Analysis

4.8.1 [324] The system shall allow the user to set the type of concentration analysis to be performed. The types of analysis known to the system are given in table 6 below.

4.8.2 [325] The system shall accept samples within the ranges given below. Results that fall outside the ranges are “Invalid”. The ranges are given in table 6 below.

4.8.3 [326] The system shall display one of the following results based on the findings: “Accepted”, “Warning”, and “Alarm”. The values for the thresholds are given in table 6 below.

Table 6: Standard Analysis Type Table

Type	Measuring range	Highest accept	Highest warning	Sample price
NCS	2 – 315 incl.	65	270	0.35 €
C2O	0.01 – 0.89 incl.	0.3	0.65	0.40 €
BHZ	0 – 9	4.5	7.5	0.75 €
LOY	100 -	200	500	0.25 €

4.8.10 [339] The system shall ensure that the format for sample numbers as presented below is correct before a sample can be processed.

The format for a sample number is:

Sample no. = T”–“n[n]”-“nnn”-“dd”.”mm“.“yy

Where:

T = “A” | “S” | “M”
n = numbers from 0 – 9
dd = numbers from 01 – 31
mm = numbers from 01 – 12
yy = numbers from 00 – 99

4.8.11 [341] The system shall accept sample numbers consisting of 4 parts divided by hyphens, namely:

- action type (A, S, or M)
- sample type (one or 2 digits)
- sample id (3 digits)
- the date the sample was taken

Examples: A-2-344-31.08.04 M-01-255-22.12.93

Explanation: When the sample is entered into the machine, the machine reads the number, but disregards the date.

The way the machine is going to examine the sample depends on the action type, in that A means automatically, S means semi-automatically, and M means manually.

The way the sample type is interpreted depends on the action type. In the case of automatic handling 1 stands for printing of a report, while 2 stands for no report. For semi-automatic handling the sample type determines how the analysis is to be performed. The sample type does not matter for manual analysis.

In the automatic handling of the sample the analysis is performed and the result stored under the sample id. If the sample id is not found in the database the analysis will not be performed and an error message will be shown telling that the sample is not registered. The steps for an automatic analysis must, of course, be found in the database as well.

To be able to run a semi-automatic analysis the sample type must be found in the database. The steps must also be found, and it must be indicated which steps could be skipped if the user chooses that. When the analysis is complete, a report is printed, including the performed and the skipped steps.

For a manual analysis the user chooses each step explicitly. The user must write the report with the results as a text, which is printed when the analysis is completed.

4.13 Lid Operation

4.13.1 [581] One of the variants of the product shall be equipped with a lid to protect the technicians performing the analyses.

The lid covers the carousel when it is moving. The lid has to be locked before the carousel is started and it is not possible to open it before the carousel has stopped completely. Two sensors are in place to detect if the lid is locked and if the carousel is moving.

As long as the lid is locked it is possible to start the carousel moving either forwards or backwards. To change the direction it is necessary to stop the carousel first, but it is not necessary to open the lid.

The manoeuvring panel has the following buttons:

‘Lock’

‘Open’

‘Forward’

‘Backward’

‘Stop’

Test Design Specification for PC part of UV/TIT-14 33a.

Version 1.0

Versions

Date	Version	Prepared by	Description of change
23-05-09	1.0	AMJ	First draft

1. Introduction

1.2 Purpose

The purpose of this part of the test specification is to provide an overview of what will be tested in the system test.

1.3 Reference documents

The following documents provide the test basis for this test design:

Ref.no.	Description	ID
[1]	System requirements, test basis.	Requirements Specification for PC part of UV/TIT-14 33a.; Vers. 1.8

1.4 Abbreviations

PCUV	PC part of UV/TIT-14 33a.
TC	Test case
UC	Use case
CRUD	Create, Read, Update, Delete
TBD	To be defined, that is it is not yet known what is to be written.

2. Feature Sets

This chapter describes the overall structure of the system test of PCUV, in that the test is divided into overall feature sets.

The following information is provided for each feature set:

(nn): Unique number that must never be changed. This is used for traceability purposes.

ns: Section or sorting number, which could be used to facilitate the reading of the document.

Description: A short description of what to test.

Approach: Description of the test design techniques to be used in the design of the test.

Traceability: Reference to the requirements in the feature set. The traceability will contain a list of unique IDs referencing requirements in [1].

This test is divided into the following feature sets:

1. (FS1) Setup of the system
2. (FS4) Calibration of UV, IR and burettes
3. (FS2) Identification of compounds
4. (FS3) Concentration of compounds (UV + control of burettes)
5. (FS6) Control of conveyor system

Feature Set (FS1): Setup of the system

Objective: To test the set-up of the system, including delivered data and reporting on calibration

Priority: Above middle

Approach: Structural testing of menus, simple requirements testing (Y/N), and equivalence partitioning and boundary value analysis

Traceability: [22], [34], [35], [36], [37], …

Feature Set (FS2): Identification of compounds

Objective: To test the identification and reporting of compounds

Priority: High

Approach: Simple requirements testing (Y/N), equivalence partitioning and boundary value analysis, syntax testing, and classification tree testing.

Traceability: [324], [325], [326], [339], [341], ….

Feature Set (FS3): TBD (not completed yet)

Feature Set (FS4): TBD (not completed yet)

Feature Set (FS5): TBD (not completed yet)

Feature Set (FS6): Control of conveyor system

Objective: To test the conveyor system, incl. speed, correct start and stop positions, lid operation etc.

Priority: Below middle

Approach: … , state-transition testing, …

Traceability: [581], …

3. Test Conditions

In this chapter the test conditions for each feature set are documented.

.
.
.

3.3 Feature Set (FS2): Identification of compounds

This feature set covers requirements related to the identification and reporting of compounds. The feature set has a number of conditions arranged in sub-sections corresponding to related requirements.

.

.

.

3.3.7 Test Conditions for Measuring Range

The test conditions for measuring range can be expressed using a simple classification tree (same as equivalence partitioning) and associated boundary value analysis. All these test conditions are traceable to the same requirement, and they have the same priority.

Requirement(s) covered: **[324-NCS], [325-NCS]**		**Priority: Above middle**	
Test condition	**Aspect**	**Sub-domains**	**Comment**
	Input		
(FS2).5.1		**< 2**	Out of range
(FS2).5.2		**2 – 315 incl.**	Valid
(FS2).5.3		**> 315**	Out of range

Requirement(s) covered: **[324-NCS], [325-NCS]**		**Priority: Above middle**	
Test condition	**boundary type**	**value**	**comment**
(FS2).5.1.a	L	unknown	Out of range
(FS2).5.1.b	U	1	Out of range
(FS2).5.2.a	L	2	Valid
(FS2).5.2.b	U	315	Valid
(FS2).5.3.a	L	316	Out of range
(FS2).5.3.b	U	unknown	Out of range

3.3.8 Test Conditions for Analysis Method

The test conditions for the identification of the analysis method from the sample number can be expressed using a classification tree. All these test conditions are traceable to the same requirement.

Requirement(s) covered: [341]									
Test condition	**aspect**	**Domain**	**aspect**	**domain**	**aspect**	**domain**	**aspect**	**domain**	**Prio.**
	action								
		A-automatic							
			sample type						
(FS2).8.1				**1-print report**					Am
(FS2).8.2				**2-no report**					Am
			sample id in database						
				yes					
					steps in database				
(FS2).8.3						**yes**			H
(FS2).8.4						**no**			H
(FS2).8.5				**no**					Am
		S-semiautomatic							
			sample type in database						
				yes					
					steps in database				
						yes			
							step [n] will be skipped		
(FS2).8.6								**yes**	Am
(FS2).8.7								**no**	Am
(FS2).8.8						**no**			H
(FS2).8.9				**no**					Am
(FS2).8.10		**M-manually**							H

.
.
.

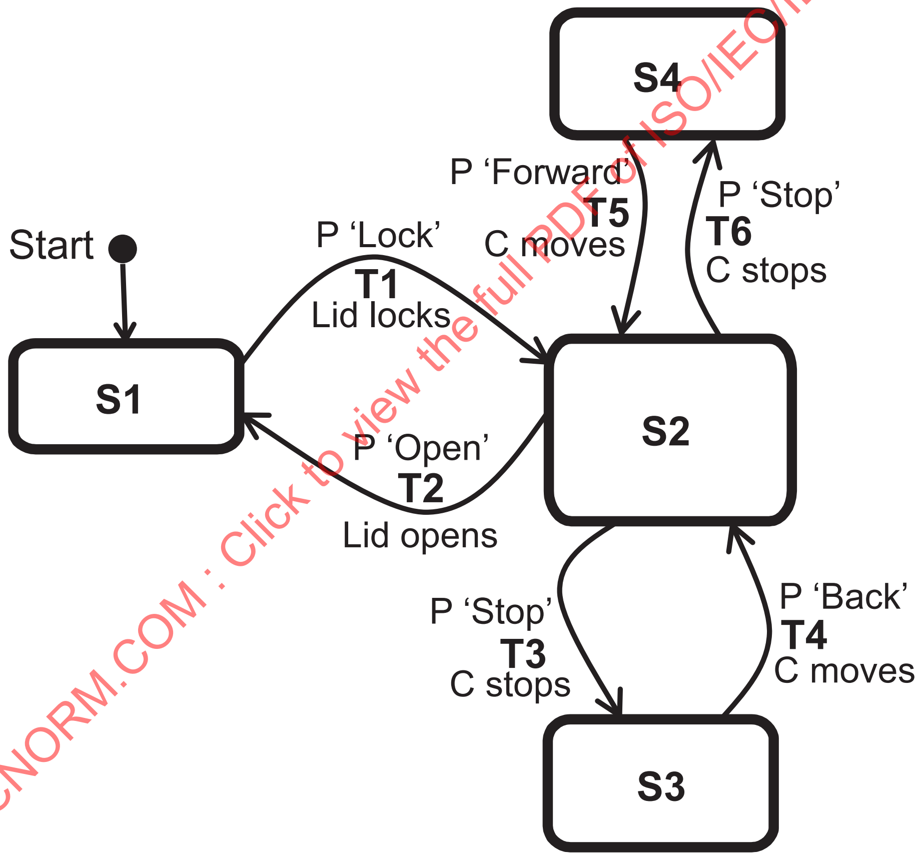
3.5 Feature Set (FS6): Control of conveyor system

This feature set covers requirements related to the conveyor system, incl. speed, correct start and stop positions, lid operation etc. The feature set has a number of conditions arranged in sub-sections corresponding to related requirements.

.
.
.

3.5.2 Test Conditions for Lid Operation

The lid operation can be illustrated in the following state-machine diagram, where the states and transitions are numbered, and the event "P" means "Press":



Requirement(s) covered: [581]		**Priority: Above middle**	
Test condition			
(FS6).11.1	The lid operation works according to the state-machine		
(FS6).11.2	All invalid (un-shown) transitions are null-transitions		

End of example

Annex J
(informative)

Test Case Specification

J.1 Example 1 – Agile Corporation

Agile Corporation is a large publication organization producing magazines and books. See more details in the introduction in Annex C.

The test coverage items and test cases for a story are summarized into test case headings and noted on the back of the story card like this:

Confirmation: **214**

1. The secretary can create a new subscription type
2. The secretary can enter name, available lengths, associated prices, and comments for a new subscription type
3. The secretary can store a new subscription type
4. The secretary can see an existing subscription type
5. The secretary can change the name, available lengths and associated prices for a subscription type as long as there is no subscription for it
6. The secretary can cancel the change of a subscription type before it is stored
7. The secretary can store the changes to a subscription type

Check: 0.1 (Annette)
a. The secretary story for "deletion" of subscription could be missing, so we should review with customer. Otherwise this is okay.

J.2 Example 2 – Traditional Ltd

Traditional Ltd is a small company that produces advanced analysis equipment to the farming industry. See more details in the introduction in Annex C.

Test Case Specification for PC part of UV/TIT-14 33a.

Version 1.0

Versions

Date	Version	Prepared by	Description of change
17-06-09	1.0	AMJ	First draft

1. Introduction

1.2 Purpose

The purpose of this part of the test specification is to list the test cases derived from the test conditions.

1.3 Reference documents

The following documents provide the test basis for this test design:

Ref.no.	Description	ID
[1]	System requirements, test basis.	Requirements Specification for PC part of UV/TIT-14 33a.; Vers. 1.8
[2]	Feature sets and test conditions	Test Design Specification for PC part of UV/TIT-14 33a.; Vers. 1.0

1.4 Abbreviations

PCUV	PC part of UV/TIT-14 33a.
TC	Test case
UC	Use case
CRUD	Create, Read, Update, Delete
TBD	To be defined, that is it is not yet known what is to be written.

2. Test Coverage Items

This section describes the coverage items that can be derived from the test conditions provided in [2].

.

.

.

2.3 Feature Set (FS2): Identification of compounds

This feature set covers requirements related to the identification and reporting of compounds. The feature set has a number of coverage items derived from the test conditions; these are arranged in sub-sections corresponding to related requirements.

.
.
.

2.3.7 Coverage Items for Measuring Range

There are three valid equivalence partitions and six valid boundaries (of which two are unknown: one below 0 and one greater than 315).

The coverage items can be reduced to be the test conditions:

(FS2).5.2, (FS2).5.1.b, (FS2).5.2.a, (FS2).5.2.b, (FS2).5.3.a,

found in section 3.3.7 in [2].

2.3.8 Coverage Items for Analysis Method

The coverage items are the leaves in the classification tree in section 3.3.8 in [2], i.e. the sub-domains written in bold.

There are 10 valid leaves (coverage items).

.

.

.

2.5 Feature Set (FS6): Control of conveyor system

.

.

.

2.5.2 Coverage Items for Lid Operation

To get Chow's 0-switch coverage of the state-machine in (FS6).11.1 in [2] there are the following 6 transitions (coverage items):

	CI1	**CI2**	**CI3**	**CI4**	**CI5**	**CI6**
SS (TC)	S1	S2	S4	S2	S3	S2
Input	P 'L'	P 'F'	P 'S'	P 'B'	P 'S'	P 'O'
Exp. output	L l	C m f	C s	C m b	C s	L o
ES (TC)	S2	S4	S2	S3	S2	S1

The null-transitions are identified in this table, shown in bold italic.

	P 'L'	**P 'F'**	**P 'S'**	**P 'B'**	**P 'O'**
S1	S2/ L l	***S1/N***	***S1/N***	***S1/N***	***S1/N***
S2	***S2/N***	S4/C m f	***S2/N***	S3/C m b	S1/L o
S3	***S3/N***	***S3/N***	S/C s	•	•
S4	***S4/N***	***S4/N***	S/C s	•	•

There are 14 null-transitions (coverage items).

.

.

.

3. Test Cases

This section describes the test cases derived from the test coverage items listed above.

3.3 Feature Set (FS2): Identification of compounds

.

.

.

3.3.7 Measuring Range

Test Case ID: 17–1 **Priority: Am** Tracing: (FS2).5.1.b	**Purpose:** to test the reaction to a sample value that is out of range
Preconditions:	The apparatus must be ready for sampling analysis. An NCS sample that has a value of 1 must be prepared.
Input:	Insert the sample and start the analysis.
Expected result:	The display shows “Invalid sample”.

.

.

.

Test Case ID: 17–4 **Priority: Am** Tracing: (FS2).5.2.b	**Purpose:** to test the reaction to a sample value that is on the upper boundary of a valid sample.
Preconditions:	The apparatus must be ready for sampling analysis. An NCS sample that has a value of 315 must be prepared.
Input:	Insert the sample and start the analysis.
Expected result:	The display shows “Warning”.

.

.

.

3.3.8 Analysis Method

.

.

.

Test Case ID: 21–3 **Priority: Am** Tracing: (FS2).8.1	**Purpose:** to test an automatic analysis of type 1.
Preconditions:	The database must include: A sample type “1” with appropriate steps; A sample id of “314”. The form where the sample id is entered must be current.
Input:	Enter a sample with sample id: A-1-314-221204