

AS/NZS ISO/IEC/IEEE 29119.3:2022
ISO/IEC/IEEE 29119-3:2021



Australian/New Zealand Standard™

Software and systems engineering — Software Testing

Part 3: Test Documentation



AS/NZS ISO/IEC/IEEE 29119.3:2022

This joint Australian/New Zealand Standard™ was prepared by joint Technical Committee IT-015, Software and Systems Engineering. It was approved on behalf of the Council of Standards Australia on 12 May 2022 and by the New Zealand Standards Approval Board on 4 May 2022.

This Standard was published on 27 May 2022.

The following are represented on Committee IT-015:

- Australian Computer Society
- Australian Digital Health Agency
- Australian Society for Technical Communication (NSW)
- Department of Defence (Australian Government)
- Engineers Australia
- Griffith University
- Institute of IT Professionals New Zealand
- IT Service Management Forum Australia
- NSW Business Chamber
- Systems Engineering Society of Australia
- Universities New Zealand
- University of New South Wales
- University of Southern Queensland
- University of Technology Sydney

This Standard was issued in draft form for comment as DR AS/NZS ISO/IEC/IEEE 29119.3:2022.

Keeping Standards up-to-date

Ensure you have the latest versions of our publications and keep up-to-date about Amendments, Rulings, Withdrawals, and new projects by visiting:

www.standards.org.au

www.standards.govt.nz

ISBN 978 1 76113 770 9

Australian/New Zealand Standard™

Software and systems engineering — Software Testing

Part 3: Test Documentation

Originated as AS/NZS ISO/IEC/IEEE 29119.3:2015.
Second edition 2022.

COPYRIGHT

© ISO/IEC/IEEE 2022 — All rights reserved

© Standards Australia Limited/the Crown in right of New Zealand, administered by the New Zealand Standards Executive 2022

All rights are reserved. No part of this work may be reproduced or copied in any form or by any means, electronic or mechanical, including photocopying, without the written permission of the publisher, unless otherwise permitted under the Copyright Act 1968 (Cth) or the Copyright Act 1994 (New Zealand).

Preface

This Standard was prepared by the joint Standards Australia/Standards New Zealand Committee IT-015, Software and Systems Engineering, to supersede AS/NZS ISO/IEC/IEEE 29119.3:2015.

The objective of this document is to specify software test documentation templates that can be used for any organization, project or testing activity. It describes the test documentation that is an output of the processes specified in AS/NZS ISO/IEC/IEEE 29119.2:2022.

This document applies to testing in all software development lifecycle models. It is intended for testers, test managers, developers, and project managers, particularly those responsible for governing, managing, and implementing software testing.

This document is identical with, and has been reproduced from, ISO/IEC/IEEE 29119-3:2021, *Software and systems engineering — Software testing — Part 3: Test documentation*

As this document has been reproduced from an International document, a full point substitutes for a comma when referring to a decimal marker.

Australian or Australian/New Zealand Standards that are identical adoptions of international normative references may be used interchangeably. Refer to the online catalogue for information on specific Standards.

The terms “normative” and “informative” are used in Standards to define the application of the appendices or annexes to which they apply. A “normative” appendix or annex is an integral part of a Standard, whereas an “informative” appendix or annex is only for information and guidance.

Contents

Preface	ii
Foreword	vii
Introduction	viii
1 Scope	1
2 Normative references	1
3 Terms and definitions	1
4 Conformance	4
4.1 Intended usage	4
4.1.1 General	4
4.1.2 Full conformance	5
4.1.3 Tailored conformance	5
5 Common information for all test documentation	6
5.1 Overview	6
5.2 Common information elements	6
5.2.1 Unique identifier	6
5.2.2 Issuing organization	6
5.2.3 Approval authority	6
5.2.4 Change history	6
5.2.5 Status	6
5.2.6 Introduction	6
5.2.7 Scope	6
5.2.8 References	7
5.2.9 Glossary	7
6 Organizational test process documentation	7
6.1 Overview	7
6.2 Test policy	7
6.2.1 Overview	7
6.2.2 Objectives of testing	7
6.2.3 Test process	7
6.2.4 Test organization structure	8
6.2.5 Tester training	8
6.2.6 Tester ethics	8
6.2.7 Standards	8
6.2.8 Other relevant policies	8
6.2.9 Measuring the value of testing	8
6.2.10 Test asset archiving and reuse	8
6.2.11 Test process improvement	8
6.3 Organizational test practices	8
6.3.1 Overview	8
6.3.2 Organization-level test practice statements	9
6.3.3 Test level/type-specific organizational test practice statements	10
7 Test management processes documentation	12
7.1 Overview	12
7.2 Test plan	12
7.2.1 Overview	12
7.2.2 Context of testing	12
7.2.3 Assumptions and constraints	13
7.2.4 Stakeholders	13
7.2.5 Testing communication	13
7.2.6 Risk register	13
7.2.7 Test strategy	14

7.2.8	Testing activities and estimates	16
7.2.9	Staffing	16
7.2.10	Schedule	17
7.3	Test status report	17
7.3.1	Overview	17
7.3.2	Test status	17
7.3.3	Reporting period	18
7.3.4	Progress against test plan	18
7.3.5	Factors blocking progress	18
7.3.6	Test measures	18
7.3.7	New and changed risks	18
7.3.8	Planned testing	18
7.4	Test completion report	18
7.4.1	Overview	18
7.4.2	Summary of testing performed	18
7.4.3	Deviations from planned testing	18
7.4.4	Test completion evaluation	19
7.4.5	Factors that blocked progress	19
7.4.6	Test measures	19
7.4.7	Residual risks	19
7.4.8	Test deliverables	19
7.4.9	Reusable test assets	19
7.4.10	Lessons learned	19
8	Dynamic test processes documentation	19
8.1	Overview	19
8.2	Test model specification	20
8.2.1	Overview	20
8.2.2	Unique identifier	20
8.2.3	Objective	20
8.2.4	Priority	21
8.2.5	Test strategy extract	21
8.2.6	Test model	21
8.2.7	Traceability	21
8.3	Test case specification	21
8.3.1	Overview	21
8.3.2	Test coverage items	21
8.3.3	Test cases	22
8.4	Test procedure specification	23
8.4.1	Overview	23
8.4.2	Unique identifier	24
8.4.3	Objective	24
8.4.4	Priority	24
8.4.5	Start up	24
8.4.6	Ordered test cases	24
8.4.7	Relationship to other procedures	24
8.4.8	Stop and wrap up	24
8.5	Test data requirements	24
8.5.1	Overview	24
8.5.2	Unique identifier	25
8.5.3	Description	25
8.5.4	Responsibility	25
8.5.5	Period needed	25
8.5.6	Resetting needs	25
8.5.7	Archiving or disposal	25
8.6	Test environment requirements	25
8.6.1	Overview	25
8.6.2	Unique identifier	26
8.6.3	Test environment item	26

8.6.4	Description.....	26
8.6.5	Responsibility.....	26
8.6.6	Period needed.....	26
8.7	Test data readiness report.....	27
8.7.1	Overview.....	27
8.7.2	Unique identifier.....	27
8.7.3	Description of status.....	27
8.8	Test environment readiness report.....	27
8.8.1	Overview.....	27
8.8.2	Unique identifier.....	27
8.8.3	Description of status.....	27
8.9	Actual results and test result.....	27
8.9.1	General.....	27
8.9.2	Actual results.....	28
8.9.3	Test result.....	28
8.10	Test execution log.....	28
8.10.1	Overview.....	28
8.10.2	Unique identifier.....	29
8.10.3	Date/time.....	29
8.10.4	Description.....	29
8.10.5	Impact.....	29
8.11	Test incident report.....	29
8.11.1	Overview.....	29
8.11.2	Timing information.....	29
8.11.3	Originator.....	29
8.11.4	Context.....	30
8.11.5	Description of the incident.....	30
8.11.6	Originator's assessment of severity.....	30
8.11.7	Originator's assessment of priority.....	30
8.11.8	Risk.....	30
8.11.9	Status of the incident.....	30
Annex A	(normative) ISO/IEC/IEEE 29119-3 requirements, recommendations and permissions.....	31
Annex B	(informative) Overview of examples.....	36
Annex C	(informative) Test policy.....	38
Annex D	(informative) Organizational test practices.....	40
Annex E	(informative) Test plan.....	46
Annex F	(informative) Test status report.....	54
Annex G	(informative) Test completion report.....	58
Annex H	(informative) Test model specification.....	60
Annex I	(informative) Test case specification.....	62
Annex J	(informative) Test procedure specification.....	64
Annex K	(informative) Test data requirements.....	66
Annex L	(informative) Test environment requirements.....	67
Annex M	(informative) Test data readiness report.....	68
Annex N	(informative) Test environment readiness report.....	69
Annex O	(informative) Actual results.....	70
Annex P	(informative) Test result.....	71
Annex Q	(informative) Test execution log.....	73
Annex R	(informative) Incident report.....	74

Annex S (informative) **Mappings to IEEE 829:2008 (superseded)** **76**

Annex T (informative) **Test models** **82**

Bibliography **83**

IEEE Notices and Abstract **85**

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.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of ISO/IEC documents should be noted. This document was drafted in accordance with the rules given in the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

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.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see <https://patents.iec.ch>).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT), see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

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 Systems and Software Engineering Standards Committee of the IEEE Computer Society, under the Partner Standards Development Organization cooperation agreement between ISO and IEEE.

This second edition cancels and replaces the first edition (ISO/IEC/IEEE 29119-3:2013), which has been technically revised.

The main changes compared to the previous edition are as follows:

- the concept of test conditions has been replaced by test models, as feedback on the previous edition of this document highlighted a problem with users' understanding of 'test conditions' and their use for deriving test cases.

A list of all parts in the ISO/IEC/IEEE 29119 series can be found on the ISO and IEC websites.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html and www.iec.ch/national-committees.

Introduction

The purpose of ISO/IEC/IEEE 29119 (all parts) 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.

ISO/IEC/IEEE 29119-1 introduces software testing concepts. This document uses the concepts of ISO/IEC/IEEE 29119-1.

ISO/IEC/IEEE 29119-2 comprises test process descriptions that define the software testing processes at the organisational level, test management level and dynamic test levels. It supports dynamic testing, functional and non-functional testing, manual and automated testing and scripted and unscripted testing, and can be utilized within any lifecycle model, including agile and traditional methodologies. Supporting diagrams describing the processes are also provided.

ISO/IEC/IEEE 29119-4 defines software test design techniques, which can be used within any lifecycle and for any product.

ISO/IEC/IEEE 29119-5 addresses the use of keyword-driven testing.

This document defines templates and provides examples of test documentation that are produced during the test process. An overview of the test documentation is provided in [Figure 1](#). The templates are arranged within clauses reflecting the overall test process description structure in ISO/IEC/IEEE 29119-2, i.e. by the test process in which they are being produced. [Annex A](#) contains a list of all the information items identified in [Clauses 6, 7 and 8](#) with the corresponding level of conformance (shall/should/may) from ISO/IEC/IEEE 29119-2. [Annex B](#) contains an overview of the examples. [Annexes C to R](#) contain examples of the application of the templates. [Annex S](#) provides mappings to existing standards. [Annex T](#) explains why the concept of test conditions has been replaced by test models in this document. A Bibliography is provided at the end of the document.

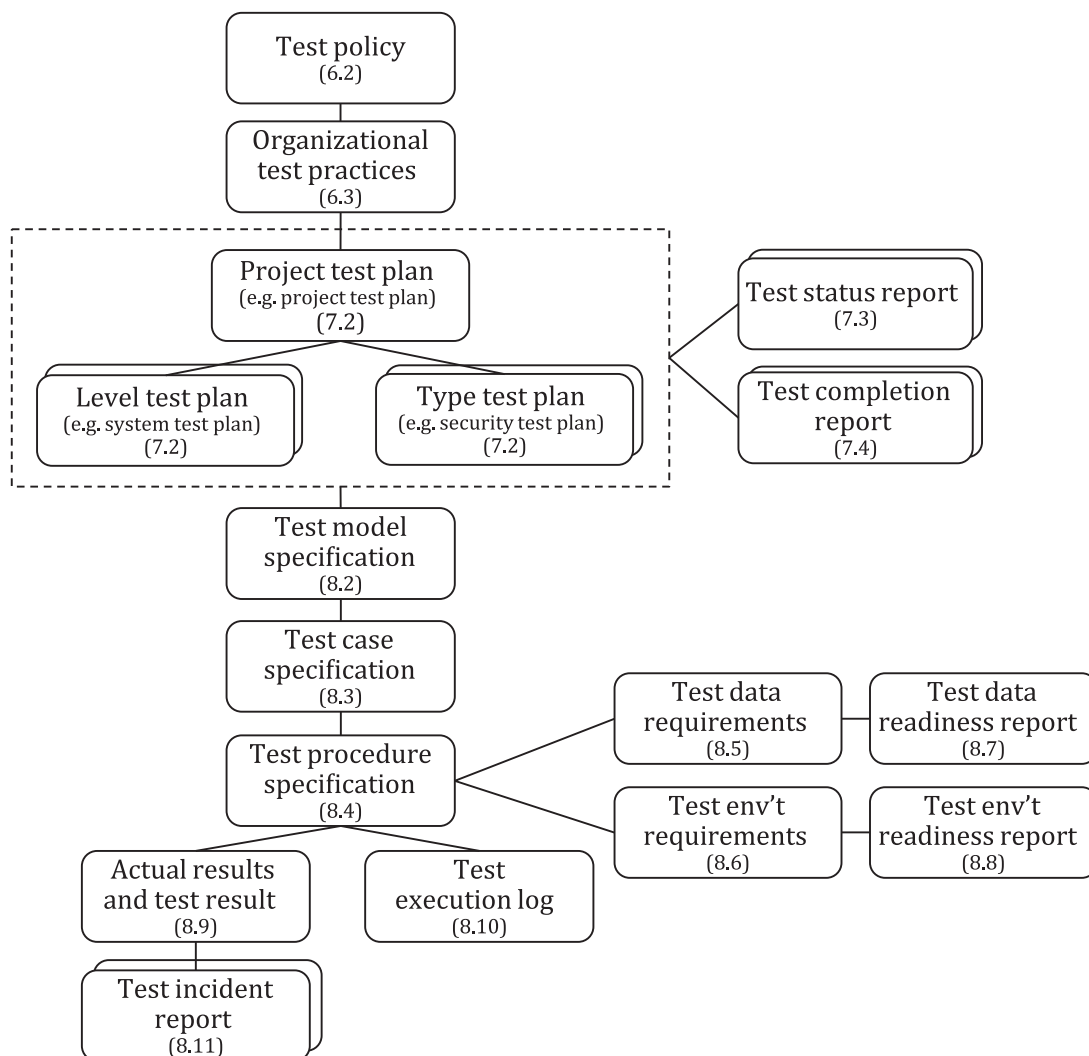


Figure 1 — Overview of test documentation

The test documentation described in this document can be on paper or in electronic form (e.g. records in test tools, spreadsheets, mind maps, white board photos).

The nomenclature of test documentation within this document (e.g. document names, section headings) and the contents of each document can be tailored to suit the unique needs of an organization, under the tailoring clause (see [Clause 4](#)).

This document uses the traditional concept of organizations and projects, but some organizations, especially those using an agile approach, do not organize their development in terms of projects; instead they run product development based on more long-lasting product teams. Users of this document can substitute the term 'product' for 'project', where appropriate.

ISO/IEC/IEEE 29119 (all parts) aims to provide stakeholders with the ability to manage and perform software testing in any organization. This document can be adopted under any lifecycle methodology including traditional (e.g. waterfall, iterative), agile or DevOps.

NOTES

Australian/New Zealand Standard

Software and systems engineering — Software Testing

Part 3: Test Documentation

1 Scope

This document specifies software test documentation templates that can be used for any organization, project or testing activity. It describes the test documentation that is an output of the processes specified in ISO/IEC/IEEE 29119-2.

This document is applicable to testing in all software development lifecycle models. This document is intended for, but not limited to, testers, test managers, developers, and project managers, particularly those responsible for governing, managing, and implementing software testing.

2 Normative references

There are no normative references in this document.

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO, IEC and IEEE maintain terminology databases for use in standardization at the following addresses:

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <https://www.electropedia.org/>
- IEEE Standards Dictionary Online: available at <https://ieeexplore.ieee.org/xpls/dictionary.jsp>

NOTE For additional terms and definitions in the field of systems and software engineering, see ISO/IEC/IEEE 24765, which is published periodically as a “snapshot” of the SEVOCAB (Systems and software Engineering Vocabulary) database and is publicly accessible at <https://www.computer.org/sevocab>.

3.1

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.

3.2

expected results

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

3.3

incident

anomalous or unexpected event, set of events, condition, or situation at any time during the life cycle of a project, product, service, or system

3.4

incident report

documentation of the occurrence, nature, and status of an *incident* (3.3)

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.

3.5

organizational test practices

documentation that expresses the recommended approaches or methods for the testing to be performed within an organization, providing detail on how the testing is to be performed

Note 1 to entry: The organizational test practices is aligned with the *test policy* ([3.20](#)).

Note 2 to entry: An organization can have more than one organizational test practices document to cover markedly different contexts, such one for mobile apps and one for safety-critical systems.

Note 3 to entry: The organizational test practices can incorporate the context of the test policy where no separate test policy is available.

3.6

organizational test specification

documentation 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 *test policy* ([3.20](#)) and *organizational test practices* ([3.5](#)).

3.7

test basis

information used as the basis for designing and implementing test cases

Note 1 to entry: The test basis can take the form of documentation, such as a requirements specification, design specification, or module specification, but can also be an undocumented understanding of the required behaviour.

3.8

test case specification

documentation of a set of one or more test cases

3.9

test completion report

test summary report

report that provides a summary of the testing that was performed

3.10

test data readiness report

documentation describing the status of each test data requirement

3.11

test environment item

element of a test environment that can be considered separately from other parts of the test environment

EXAMPLE Hardware, software, interfaces, peripherals, tools.

3.12

test environment readiness report

documentation that describes the status of each test environment requirement

Note 1 to entry: This can list the status of each of the *test environment requirements* ([3.13](#)).

3.13

test environment requirements

documentation of the necessary properties of the test environment

Note 1 to entry: All or parts of the test environment requirements can reference where the information can be found, e.g. in the appropriate *organizational test practices* ([3.5](#)) document, *test plan* ([3.19](#)) and/or *test specification* ([3.23](#)).

3.14**test execution log**

record of the execution of one or more test procedures

3.15**test incident**

event occurring during the execution of a test that requires investigation

3.16**test model**

representation of the test item, which allows the testing to be focused on particular characteristics or qualities

EXAMPLE Requirements statements, equivalence partitions, state transition diagram, use case description, decision table, input syntax, source code, control flow graph, parameters and values, classification tree, natural language.

Note 1 to entry: The test model and the required test coverage are used to identify test coverage items.

Note 2 to entry: A separate test model can be required for each different type of required test coverage included in the test completion criteria.

Note 3 to entry: A test model can include one or more test conditions.

Note 4 to entry: Test models are commonly used to support test design (e.g. they are used to support the test design in ISO/IEC/IEEE 29119-4, and they are used in model-based testing). Other types of models exist to support other aspects of testing, such as test environment models, test maturity models and test architecture models.

3.17**test model specification**

documentation specifying the *test model* ([3.16](#))

3.18**test organization**

management structure responsible for testing within an organization

Note 1 to entry: The test organization is typically technically, managerially and financially independent from the development organization.

3.19**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 can have more than one test plan, for example there can 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 can be defined in one or more test level / test type plans (e.g. a system test plan or a performance test plan).

Note 2 to entry: A test plan is typically a written document, although other formats can be possible as defined locally within an organization or project.

Note 3 to entry: Test plans can also be written for non-project activities, for example a maintenance test plan.

3.20**test policy****organizational test policy**

executive-level documentation 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 organization's testing.

3.21

test procedure specification

test script

documentation specifying one or more test procedures

3.22

test result

indication of whether or not a specific test case has passed or failed, i.e. if the *actual results* (3.1) corresponds to the *expected results* (3.2) or if deviations were observed

3.23

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 can be detailed in one document, in a set of documents, or in other ways, for example in a mixture of documents and database entries.

3.24

test status report

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

3.25

test strategy

part of the *test plan* (3.19) that describes the approach to testing for a specific project, test level or test type

Note 1 to entry: The test strategy usually describes some or all of the following; the test levels and test types 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.

3.26

test traceability matrix

verification cross reference matrix

requirements test matrix

requirements verification table

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

Note 1 to entry: Different test traceability matrices can have different information, formats, and levels of detail.

4 Conformance

4.1 Intended usage

4.1.1 General

The requirements in this document are contained in [Clauses 4, 5, 6, 7, 8](#) and [Annex A](#).

This document provides requirements for test documentation across the complete software testing lifecycle. Test documentation shall be considered as conforming if it is available in an electronic format (e.g. records in a test management tool or spreadsheet), divided into separate documents, or combined with other documents into one document.

In this document, for simplicity, each item of test documentation is described as if it were a separate hardcopy document. Test documentation titles, headings and layout described in this document may be

modified (e.g. added to, combined or re-titled). The use of the nomenclature used in [Clauses 5, 6, 7 and 8](#) is not mandatory.

The information items identified in [Clauses 6, 7 and 8](#) correspond to the outputs of the test processes in ISO/IEC/IEEE 29119-2. [Annex A](#) provides an overview of the requirements for clauses in ISO/IEC/IEEE 29119-2 where the creation of the information items defined in [Clauses 6, 7 and 8](#) are described.

It is recognized that some organizations, projects or teams may not need to use all of the test documentation defined by this document. Therefore, implementation of this document typically involves selecting a set of test documentation suitable for the organization, project or team. There are two ways that an implementation can be claimed to conform to the provisions of this document: full conformance and tailored conformance.

The organization shall assert whether it is claiming full or tailored conformance to this document.

NOTE 1 Options for conformance are provided for needed flexibility in the application of this document.

NOTE 2 When this document is used to help develop an agreement between an acquirer and a supplier, clauses of this document can be selected for incorporation in the agreement with or without modification. In this case, it is more appropriate for the acquirer and supplier to claim compliance with the agreement than conformance with this document.

NOTE 3 An organization (for example, national, industrial association, company) imposing this document, as a condition of trade, can specify and make public the minimum set of test documentation, which constitute suppliers' compliance with the conditions of trade.

NOTE 4 Requirements of this document are marked by the use of the verb "shall". Recommendations are marked by the use of the verb "should". Permissions are marked by the use of the verb "may". However, despite the verb that is used, the requirements for conformance are selected as described previously.

4.1.2 Full conformance

Full conformance shall be achieved by providing evidence that all requirements (i.e. shall statements) of the test documentation defined in [Clauses 5 to 8](#) and [Annex A](#) have been met.

4.1.3 Tailored conformance

When this document is used as a basis for establishing a set of test documentation that does not qualify for full conformance, the subset of test documentation for which tailored conformance is claimed, shall be recorded. Tailored conformance is achieved by demonstrating that all requirements (i.e. shall statements) for the recorded subset of test documentation defined in [Clauses 5 to 8](#) and [Annex A](#) have been satisfied.

Where tailoring occurs, justification shall be provided (either directly or by reference), whenever the requirements (i.e. shall statements) of test documentation defined in [Clauses 5 to 8](#) and [Annex A](#) are not met. 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.

EXAMPLE Where organizations follow information item management processes in standards such as ISO 15489-1 or ISO 9001 or use similar internal organizational processes, they can decide to use information items produced by those processes in place of the test documentation defined in this document.

5 Common information for all test documentation

5.1 Overview

This clause defines the common information elements that shall be captured for all test documentation types covered in [Clauses 6, 7 and 8](#). The common information elements listed in [5.2](#) shall be included in each applicable piece of test documentation.

NOTE The test documentation described in this document can be issued in several versions over time. Whilst a simple level of version control information is defined in [Clause 5](#), the overall handling of multiple versions of test documentation (including reviewing and maintaining test documentation) is out of scope of this document, as that is a configuration management issue that is handled in other ISO standards.

5.2 Common information elements

5.2.1 Unique identifier

Uniquely identifies the test documentation or record.

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

5.2.2 Issuing organization

Specifies the organization responsible for preparing and releasing the test documentation or record. It may also include authors.

5.2.3 Approval authority

Identifies the designated persons who have the responsibility for signing off on the test documentation or record (possibly electronically). It may also include reviewers and pertinent managers.

5.2.4 Change history

Includes a log of all of changes made to the test documentation or record since its inception.

EXAMPLE This can consist of a table that includes the present version of the test documentation and any predecessor versions, with the unique identifier of each version, a description of documentation changes with respect to the previous version, the name and role of the person making the changes and reasons for changes. Reasons for change can include addressing audit comment, team review comments or system changes. The person making the change can include the test documentation author, project manager or system owner.

5.2.5 Status

Defines the status of each item of test documentation.

EXAMPLE Test documentation status can include "not yet started", "in design", "in review" and "approved."

5.2.6 Introduction

Explains the context and structure of the test documentation or record.

5.2.7 Scope

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

5.2.8 References

Lists other referenced documents or records 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 can be policies, plans, procedures, and other source data.

5.2.9 Glossary

Provides a lexicon for the terms, abbreviations, and acronyms, if any, used in the test documentation or record.

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

6 Organizational test process documentation

6.1 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 practices.

The full templates with explanatory text for these items of test documentation can be found in [6.2](#) and [6.3](#). [Annex A](#) provides an abbreviated overview of each item of test documentation along with the requirements, recommendations and permissions (shall, should and may) for each section of the test documentation. [Annexes C](#) and [D](#) provide examples of a test policy and organizational test practices for example projects.

6.2 Test policy

6.2.1 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.

[A.3](#) provides an outline of the organizational test policy, while [C.1](#) and [C.2](#) provide examples that demonstrate how Organizational Test Policies can be developed for two different example projects.

The contents of the test policy are specified in [6.2.2](#) to [6.2.11](#).

6.2.2 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.

6.2.3 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 can be a test process in ISO/IEC/IEEE 29119-2. The details of activities in the test process can be described in more detailed test process documentation.

6.2.4 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.

6.2.5 Tester training

States required training and certifications for individuals performing testing in the organization.

6.2.6 Tester ethics

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

6.2.7 Standards

States which standards are applicable to the testing performed within the organization.

6.2.8 Other relevant policies

Identifies policies that impact the testing performed within the organization.

EXAMPLE A policy statement can be that testing will conform to the quality policy.

6.2.9 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.

6.2.10 Test asset archiving and reuse

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

6.2.11 Test process improvement

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

6.3 Organizational test practices

6.3.1 Overview

The organizational test practices is technical test documentation 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, where one exists.

The organizational test practices provides guidelines to projects within its scope; it is not project specific.

For small or highly homogenous organizations, a single set of organizational test practices documentation may cover all testing activities. An organization may have more than one set of organizational test practices documentation if the organization performs testing in a number of significantly different ways, such as for both safety-critical products and non-safety-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 documented test practices.

The organizational test practices may incorporate the content of the test policy where no separate test policy is available.

The practices are defined for the specified scope – such as at the level of a programme, project or for a specific level or type of testing.

The organizational test practices includes identification of test levels expected to be used and types and practice statements for each of these. It may be partitioned with a subsection for each identified test level and test type if the test level or test type practice statements differ significantly between each test level or type; this is illustrated in [Figure 2](#).

The contents of the organizational test practices are defined in [6.3.2](#) and [6.3.3](#). [A.3](#) provides an outline of the organizational test practices documentation, while [Annex D](#) provides examples that demonstrate how organizational test practices can be developed for two example organizations.

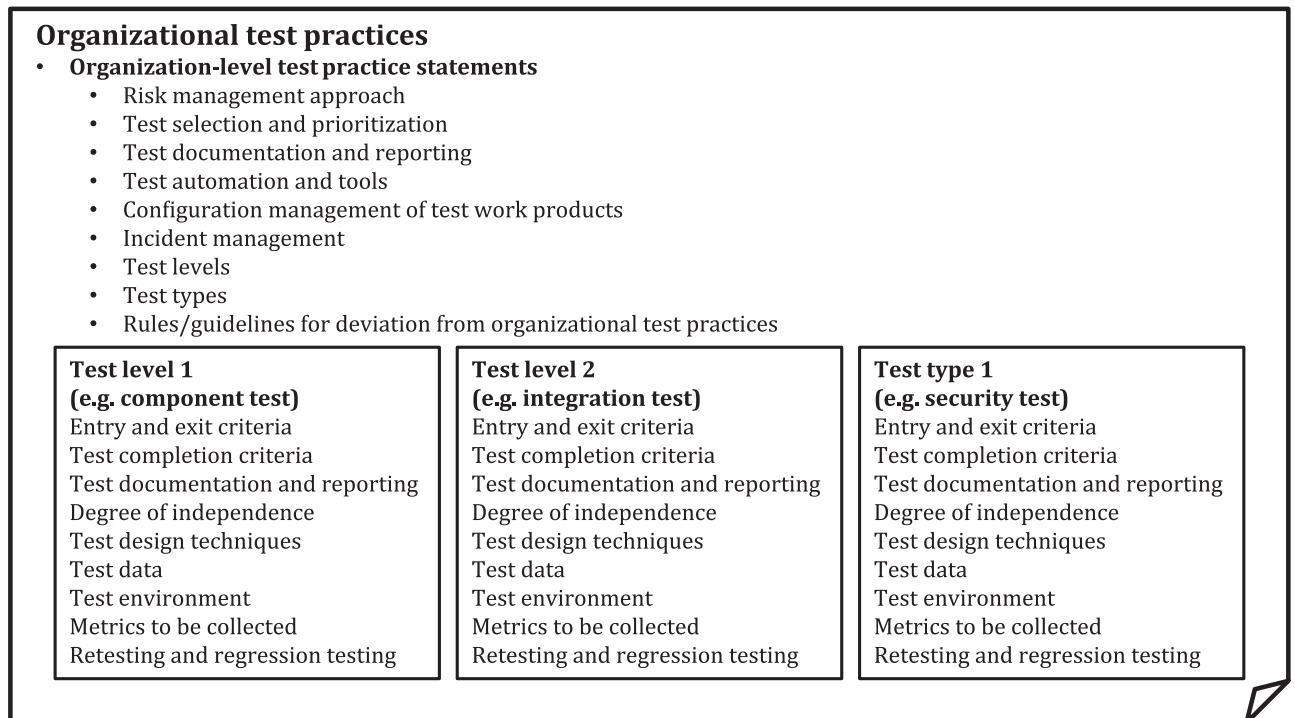


Figure 2 — An example structure of the organizational test practices document

6.3.2 Organization-level test practice statements

6.3.2.1 General

This section includes statements that are apply to all of the test levels and test types to be performed in projects run within the organization (or that part of the organization that is within the scope of the organizational test practices documentation). This section may include subsections from the test policy, if needed.

6.3.2.2 Risk management approach

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

EXAMPLE This can be to identify project and product risks that can be mitigated or reduced by testing.

6.3.2.3 Test selection and prioritization

Describes the approach to selecting and prioritizing test execution, in the form of prioritized test procedures. Test procedures consist of prioritized test cases, derived from prioritized test coverage items.

6.3.2.4 Test documentation and reporting

Identifies the test documentation expected to be produced during testing by each project being run within the organization. Describes when each documentation type is prepared and the associated approval process. This is tightly connected to the test process that is specified in the test policy.

6.3.2.5 Test automation and tools

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

EXAMPLE This can include test management tools, test execution tools, performance test tools, security test tools and usability test tools.

6.3.2.6 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.

6.3.2.7 Incident management

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

6.3.2.8 Test levels

Identifies specific test levels to be performed as part of the testing within the scope of the practices.

6.3.2.9 Test types

Identifies specific types of testing to be performed as part of testing within the scope of the practices.

6.3.2.10 Rules/guidelines for deviation from organizational test practices

Defines rules or guidelines for determining the conditions under which deviation from organizational test practices is possible/allowed within the organization.

EXAMPLE An organization can decide that individual projects or teams cannot change the defect severity and priority levels that are set out in the organizational test practices.

6.3.3 Test level/type-specific organizational test practice statements

6.3.3.1 Entry and exit criteria

Criteria that specify when test processes for a given test level or type can start and stop.

Each test level and type consists of the following test processes:

- test planning;
- test design and implementation;
- test environment and data management;
- test execution;
- test incident reporting.

Different entry and exit criteria may be defined for each of these test processes, or for selected ones, or for the entire test level or type as a whole.

EXAMPLE Entry criteria for test execution during system testing can include: the system test plan has been created and signed off, test cases have been created and approved by business stakeholders and test data and the test environment (including deployment of the test item) are prepared and ready for execution. Exit criteria for system testing can include: all high priority test cases have been executed and passed testing or that any severity 1 or 2 defects that remain open have been reviewed and approved by the business as not requiring addressing during system testing.

6.3.3.2 Test completion criteria

Criteria that specify when test execution for a given test level, test type or test technique can be considered complete. For test types and test levels, the test completion criteria may be the same as the exit criteria for that test level.

EXAMPLE The test completion criterion for state transition testing can be that test execution will continue until all states in the state model have been "visited" during testing.

6.3.3.3 Test documentation and reporting

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

6.3.3.4 Degree of independence

Establishes the level of independence of those performing the testing. States how testers or groups of testers are technically, managerially, and financially independent.

6.3.3.5 Test design techniques

Identifies specific test design techniques to be used during test design and implementation within the test level or type.

6.3.3.6 Test data

Identifies the test data required for the test level or type. This may include a statement of the origin of test data, where particular types of test data are located and groups or organizations responsible for creating and maintaining test data.

NOTE The data quality model defined in ISO/IEC 25012 can support the identification of test data requirements (and potentially data measurement needs).

6.3.3.7 Test environment

Identifies the test environment for the test level or type. This may include where specific types of test should be performed and may identify groups or organizations responsible for the test environment.

6.3.3.8 Metrics to be collected

Describes the metrics for which values are to be collected during the test activities in the test level or type.

6.3.3.9 Retesting

Identifies the practices, conditions and activities for retesting in each test level or type.

6.3.3.10 Regression testing

Identifies the practices, conditions and activities for regression testing in each test level or type.

7 Test management processes documentation

7.1 Overview

The test documentation 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 each of these items of test documentation are specified in [7.2](#) to [7.4](#). [Annex A](#) provides an abbreviated overview of each item of test documentation along with the requirements, recommendations and permissions (shall, should and may) for each section of the test documentation. [Annexes E, F and G](#) provide examples of test plans, test status reports, and test completion reports for example projects.

7.2 Test plan

7.2.1 Overview

The test plan is an item of test documentation that is used for test planning and test management. It describes the decisions made during initial test planning and evolves as re-planning is performed as part of the control activity. 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 (i.e. project test plan/master test plan), or to a specific test level or type (e.g. system test plan, integration software test plan, subsystem test plan, subcontractor software test plan, unit software test plan, or performance test plan), or to a specific iteration of testing. If multiple software test plans are created, a mapping tree may be produced to aid the documentation of relationships and the information contained in each.

If the organization does not yet have organizational test practices defined, then there may be sections from that test documentation that need to be included in the test plan. This may include the test selection and prioritization ([6.3.2.3](#)), test automation and tools ([6.3.2.5](#)), configuration management of test work products ([6.3.2.6](#)) and incident management ([6.3.2.7](#)), as these are not repeated in the test plan template defined in this subclause.

[A.3](#) provides an outline of the test plan, while [Annex E](#) provides examples that demonstrate how test plans can be developed for two different example projects.

The contents of the test plan are specified in [7.2.2](#) to [7.2.10](#).

7.2.2 Context of testing

7.2.2.1 Projects / test levels / test types

Identifies the projects, test levels and/or test types for which the test plan is being written and other relevant contextual information.

7.2.2.2 Test items

Identifies the test items for the testing covered by this test plan including version/revision information or a reference for where this information can be found.

This section may describe the purpose of the test item or reference where this information can be found.

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

EXAMPLE The test item can be a software unit, a group of units or modules, a service, a subsystem, or a complete system, app or product.

7.2.2.3 Test scope

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

EXAMPLE Features to be tested can be specific functions, interfaces, business processes or attributes of the software, such as usability or performance.

7.2.2.4 Test basis

Identifies the basis for designing and implementing tests.

7.2.3 Assumptions and constraints

Describes any assumptions and constraints for the testing covered by this plan. These may include regulatory standards, requirements in the test policy and the organizational test practices, contractual requirements, project time and cost constraints, and availability of appropriately skilled staff, tools and/or environments.

7.2.4 Stakeholders

Lists the stakeholders and their relevance to the testing.

EXAMPLE 1 This can include the stakeholder with the authority for resolving issues raised as a result of the testing activities and the stakeholder with the authority for approving test products and processes.

EXAMPLE 2 This information can be captured in a RACI (responsible, accountable, consulted, informed) matrix.

7.2.5 Testing communication

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

This information may be represented visually.

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

7.2.6 Risk register

7.2.6.1 General

Lists the risks identified as part of the test planning activity. This should include any relevant risks that may be specified in the organizational test practices. Provides an exposure level for each risk based on its impact and probability. Provides recommendations to manage the risks. This section may reference a separate risk register.

EXAMPLE Recommendations to manage risk can include to eliminate, reduce, monitor or accept a risk.

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

7.2.6.2 Product risks

Identifies test-related product risks.

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

7.2.6.3 Project risks

Identifies test-related project risks.

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

7.2.7 Test strategy

7.2.7.1 General

Describes the approach to testing for the identified scope of testing of the test item, as outlined in [7.2.7.2](#) to [7.2.7.15](#). It may refer to the organizational test practices stating only its differences from it.

7.2.7.2 Test levels

For a project test plan or a type test plan, this identifies the levels of testing that will be conducted.

7.2.7.3 Test types

For a project test plan or a level test plan, this identifies the types of testing that will be conducted.

7.2.7.4 Test deliverables

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

EXAMPLE The following test documentation can be included:

- test plan;
- test model specification;
- test case specification;
- test procedure specification;
- test data readiness report;
- test environment readiness report;
- incident reports;
- test status reports;
- 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 test documentation has been combined or eliminated, then this list will be modified accordingly.

This subsection may include information on when each item of test documentation should be delivered, and name the sender and recipient (preferably by position, not name).

7.2.7.5 Test design techniques

Specifies which test design techniques are to be applied.

7.2.7.6 Entry and exit criteria

Specifies when test processes for a given test level or type can start and stop.

Each test level and type consists of the following processes:

- test planning;
- test design and implementation;
- test environment and data management;
- test execution;
- test incident reporting.

Different entry and exit criteria may be defined for each of these test processes, or for selected ones, or for the entire test level or type as a whole.

EXAMPLE Entry criteria for test execution during system testing can include: the system test plan has been created and signed off, test cases have been created and approved by business stakeholders and test data and the test environment (including deployment of the test item) are prepared and ready for execution. Exit criteria for system testing can include: all high priority test cases have been executed and passed testing or that any severity 1 or 2 defects that remain open have been reviewed and approved by the business as not requiring addressing during system testing.

7.2.7.7 Test completion criteria

Specifies when test execution for a given test level, test type or test technique can be considered complete. For test types and test levels, the test completion criteria may be the same as the exit criteria for that test level.

EXAMPLE The test completion criterion for state transition testing can be that test execution will continue until all states in the state model have been "visited" during testing.

7.2.7.8 Degree of independence

Establishes the level of independence of those performing the testing. States how the testers are technically, managerially, and financially independent.

7.2.7.9 Metrics to be collected

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

7.2.7.10 Test data requirements

Specifies all relevant test data requirements for the project or test level or test type (as appropriate).

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

These test data requirements may be deferred to the test data requirements documentation (see [8.5](#)), as applicable.

NOTE The data quality model defined in ISO/IEC 25012 can support the identification of test data requirements (and potentially data measurement needs).

7.2.7.11 Test environment requirements

Specifies the necessary and desired properties of the test environment.

EXAMPLE 1 This can include hardware, software, test tools, databases, and personnel (identifying their organizations, as appropriate).

Information regarding selection, evaluation, acquisition and support for each component part of the test environment may be included.

It may include test environment requirements for test preparation, test execution (including data capture), and any post-execution activities.

EXAMPLE 2 A post-execution activity can be data analysis.

These test environment requirements may be defined in the test environment requirements documentation (see [8.6](#)) as applicable, but this should be stated in the test plan.

7.2.7.12 Retesting

Specifies the conditions under which retesting will be performed.

7.2.7.13 Regression testing

Specifies the conditions under which regression testing will be performed.

7.2.7.14 Suspension and resumption criteria

Specifies the criteria used to suspend and resume all or a subset of the testing activities in the test plan. Identifies who has the authority to suspend and resume testing activities. Specifies the activities that need to be completed when testing resumes, which may include repeating specific testing activities.

7.2.7.15 Deviations from the organizational test practices

Records any test plan content that deviates from the organizational test practices and the reasons for deviation. Identifies those responsible for approving deviations, where applicable.

7.2.8 Testing activities and estimates

Identifies all the necessary testing activities required to implement the test strategy. The approach and expected number of iterations for retesting and regression testing as well as any dependencies between activities should be considered.

NOTE 1 The testing activities can be described in terms of a work breakdown structure or activities on an agile activity board.

Describes estimates for each identified testing activity. Additionally, where appropriate, describes the allocated testing budget and cost estimates or references where that information can be found.

NOTE 2 Allocated budget and cost estimates can be located in the project plan.

7.2.9 Staffing

7.2.9.1 General

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

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

7.2.9.2 Roles and responsibilities

Provides an overview of the primary (they are the activity leader) and secondary (they are not the leader but provide 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 items. They may be participating either full-time or part-time.

For each testing person, specify the periods when the person is required.

EXAMPLE 1 The responsible parties can 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.

EXAMPLE 2 This information can be captured in an RACI matrix.

7.2.9.3 Hiring needs

Identifies specific requirements for additional testing staff that are needed to perform the specified testing. Specifies when the staff are needed, if they should be temporary, full-time or part-time, and their desired skill set. These may be constrained by contract and business needs.

7.2.9.4 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, such as traditional classroom training, self-paced computer-based training, training over the Internet, and mentoring by more knowledgeable staff members.

7.2.10 Schedule

Identifies test milestones defined in the project schedule and test activities 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 can be quality assurance and configuration management.

7.3 Test status report

7.3.1 Overview

The test status report provides information about the status of the testing that is performed in a specific reporting period.

NOTE In agile delivery, the test status report might not be a written document. For example, test status and progress information can be shared verbally by testers at a daily stand-up meeting. When evidence of test status reporting is not recorded, it constitutes tailored conformance (see [4.1.3](#)).

[A.3](#) provides an outline of the test status report, while [Annex F](#) provides examples that demonstrate how test status reports can be developed for two different example projects.

The contents of the test status report are specified in [7.3.2](#) to [7.3.8](#).

7.3.2 Test status

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

EXAMPLE Test status can be ahead of schedule, on time, behind schedule.

7.3.3 Reporting period

Specifies the time period covered by the report.

7.3.4 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, descriptions of any remedial actions, an account of the effects, and consideration of the implications with regard to planned project objectives.

7.3.5 Factors blocking progress

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

7.3.6 Test measures

Presents the collated test measures related to the reporting period.

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

7.3.7 New and changed risks

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

7.3.8 Planned testing

Describes the planned testing for the next reporting period.

7.4 Test completion report

7.4.1 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 level or test type.

[A.3](#) provides an outline of the test completion report, while [Annex G](#) provides examples that demonstrate how test completion reports can be developed for two different example projects.

The contents of the test completion report are specified in [7.4.2](#) to [7.4.10](#).

7.4.2 Summary of testing performed

Summarizes the testing performed across the project and/or in the test levels and test types 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 can include restrictions on test environment availability.

7.4.3 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.

7.4.4 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 and their corresponding risk treatments.

7.4.5 Factors that blocked progress

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

7.4.6 Test measures

Presents the collated test measures.

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

7.4.7 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 testing and /or any new risks identified as a result of the final monitoring and closure of the test.

7.4.8 Test deliverables

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

EXAMPLE This can include the test plan, test case specifications and test procedure specifications.

7.4.9 Reusable test assets

Lists all the reusable test assets and their location.

EXAMPLE This can include test procedures and test data.

7.4.10 Lessons learned

Describes the results of the lessons learned meeting.

NOTE If the project is following an agile methodology, the lessons learned may have been collected and captured iteratively during retrospectives. Key findings that are to be brought into future test levels, types or projects will still typically be documented in the test completion report.

8 Dynamic test processes documentation

8.1 Overview

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

- test specification, divided into:
 - test model specification;
 - test case specification;
 - test procedure specification;

NOTE 1 These can be separate documents, can appear as chapters in a test specification, can be in the form of charters, or can be electronic records in a test tool, depending on the size and nature of the test project.