Testing Strategy Template

Purpose:
This MS Word document is intended for use by Customer X planning the testing of IT systems and applications.

Conventions used in this document:
This section and all parts of this document similarly enclosed by a (12.5%) grey background box are intended to provide guidance in the development of the document. It is not intended that any such grey box remain in the document.
Throughout the non-grey text of this document, there will be standard text and examples that can be used directly in divisional documents.

Document Standards:
This document was based on the following standards :
- Information Technology – Software Packages – Quality requirements and testing – Standards Australia AS/NZS 4366:1996

Before Releasing a New Version
Update the table of contents. If the version is not Draft, remove "Draft" from the page header from the MS Word sections 3 onwards (MS Word section, not the Strategy section).

MS Word Tips:
Become familiar with updating MS Word fields, including the table of contents, headers and footers and page layout.
Beware of the often undesirable effects (through misunderstanding of the Microsoft "feature", of course) of deleting across section boundaries, most particularly the effects on headers and footers, page numbering and page orientation.

<division name>

Testing Strategy

for

<Business System>

Version : <version no.>

<Month, Year>

Uncontrolled Copy

Prepared by

Document Approval

This strategy has been developed to describe the high level test planning for the <business critical system>.

Date	Author Job Title	Author Name	Signature

This strategy has been approved by the following :

When the document is complete : has been reviewed and updated, record the authorized sign-off here.

Date	Approving Agent Job Title	Approving Agent Name	Signature
	<person 1>		
	<person 2>		

Document Control

Amendment History

Suggested rules for maintaining version details:
- Use version "Draft" whenever the version has not yet been reviewed successfully
- Change "Draft" to "1.0" when the document has been reviewed and updated and is ready for initial signoff.
- Each time a new version of this document is issued for review, append a new row to the table, documenting the change details, including that it is a "Draft" version
- Change "Draft" to a new version number e.g. "1.1", when the new version has been reviewed and updated and is ready for signoff.

Example:

Version	Date	Reason for Change	Document/ Para Ref	Made By	Description of Change
1.0		Initial Release	n/a	John Bloggs	n/a
1.1		Take account for funding changes	9. Testing schedule	John Bloggs	Compacted schedule
Draft		Delays getting resources	9. Testing schedule	John Bloggs	Amended schedule

Version	Date	Reason for Change	Document/ Para Ref	Made By	Description of Change
Draft		Initial Release	n/a	<author>	n/a

Table of Contents

> Remember to update this table of contents field prior to printing a new copy after any change.

1. Introduction

1.1 Purpose

This document describes the testing strategy that will be implemented for the <business system> to ensure that it is consistent and compliant within Customer X's Enterprise Architecture environment.

1.2 Testing Milestones

The purpose of this section is to provide an overview of milestone events, which are generally the start and end dates of testing phases. Ensure that you include meaningful milestone descriptions at this point in the document, Since testing phases have not yet been described in detail. The information may be represented as a table, Gantt chart or in standard text format.

You may wish to include more detailed schedule information than that, but be careful not to include too much detail. Otherwise you will be required to update this document (according to the documentation control requirements) each time a relatively minor schedule change occurs.

Refer to Customer level milestone dates here, if relevant.

1.3 Assumptions (within the scope of this Testing Strategy)

The purpose of this section is for assumptions to be clearly stated up front. It is intended that the assumptions stated here refer to the work that is described by the Testing Strategy. Should particular assumptions not hold true, then the issues and/or risk management processes described in a later section will need to be applied to the situation.

1.4 Dependencies (outside the scope of this Testing Strategy)

The purpose of this section is for dependencies to be clearly stated up front. It is intended that the dependencies stated here, refer to items that the successful implementation of the Testing Strategy is dependent upon and are outside the control/scope of this testing strategy. Should a particular dependency not hold true, then the issues and/or risk management processes described in a later section will need to be applied to the situation.

1.5 Standards

List the standards documents that govern the content of this testing strategy.

- Information Technology – Software Packages – Quality requirements and testing – Standards Australia AS/NZS 4366:1996

1.6 References

List the details of the documents (other than the standards described in the preceding section) that are referred to either directly, or used as a basis of preparing this document,

- <reference 1>
- <reference 2>

2. Testing Scope

2.1 Business Functions

2.1.1 Business Functions In Scope

The <business system> business functions that will be tested according to this testing strategy are :

The information described by these bullet points will be used as the basis of developing one or more test plans, so should define the testing scope of each business function in sufficiently meaningful detail.

- <Business function 1>
- <Business function 2>

2.1.2 Business Functions Out of Scope

> To avoid any future misunderstanding, make sure to document any business functions that will not be tested by this testing strategy. This is especially important if the BS is dependent on interfaces. The scope boundary must be stated clearly e.g. will you be responsible for verifying that the data transmitted by the BS via an outgoing interface is correct at the interface partner's "receiving end" ?
> To illustrate that the selected dates adequately cover the testing of these rules, you may choose to use the following table layout. If you decide not to use the table, make sure that that the rule/date coverage relationship is clearly defined. Instead of the table, you may decide to include multiple tables or sub-sections here, one per business function.

3. Testing Approach

3.1 Overview

The Testing Approach describes how the testing will be structured – the order in which different types of tests will be conducted and the work required to do so

3.2 Testing Phases

In accordance with the standards, testing will be planned and executed in four sequential phases :

1. Unit Testing of single business system modules,

1. Integration Testing of a related group of business system modules,

2. System Testing of the entire business system,

3. Acceptance Testing the entire business system with live data for production readiness.

> These are the standard phases and standard phase names used in the standards documents. If your BS project uses a different number of phases or calls them different names, consider relating your local terms to these standard ones and clearly explaining them. (Refer Standards Australia Handbook 104, section 5)

Whilst the execution of each phase will certainly be conducted sequentially, test planning for one or more phases will be conducted in parallel.

One or more Test Plans will be developed for each phase of testing, as described in a following section.

CONSULTING
CLOUD

3.3 Testing Types

3.3.1 Introduction

Many different types of tests will be planned and executed. The following table illustrates which types of testing will be planned and executed in which phase. Not all types of testing are equally applicable to each testing phase. The scope of each test type will differ, depending on the particular test phase. Further detailed descriptions of each test type may be found in the Standards Australia Handbook 104 (A guide to Year 2000 compliance).

> Edit the following table to summarize the types of testing that you plan to conduct, or replace the table with text that conveys the same message. If you are not going to conduct a particular type of test, don't delete it from the document, leave it in and explain why it's not applicable, in the Comments column. It's also a good idea to include sub-sections, which describe exceptional or specific aspects of these testing types, relevant to this BCS. The testing type usually worthy of special note is Intersystem testing.

Test Type	Unit Testing	Integrat'n Testing	System Testing	Acceptance Testing	Comment
Functional Testing					Focus is on *results* of processing
Requirements Testing	X	X	X	X	Each phase will refer to different types of requirements documents.
Regression Testing	X	X	X	X	This is required in all phases to confirm that all remains functionally correct after changes have been made.
Error Handling Testing	X	X	X	X	This is required in all phases to confirm that errors continue to be prevented, recognized or corrected.
Intersystem Testing			X	X	Confirms that proper parameters and data continue to be passed between applications and that timing of functions is synchronized across applications.
Parallel Testing			X	X	Confirms that the new version of the BCS generates the same results as the older version
Structural Testing					Focus is on the BS's *readiness for implementation*.
Operations Testing			X	X	Confirms that system is ready for normal system operations.

CONSULTING
CLOUD