

# TEST STRATEGY TEMPLATE

[Type the document subtitle/type]

[Type the company name]

Version [X.X]

Contents

[**TEST STRATEGY TEMPLATE**](#_gmixe8qzd9fz) **0**

[**INTRODUCTION**](#_1fob9te) **3**

[Purpose](#_3znysh7) 3

[Document Location](#_2et92p0) 3

[Revision History](#_tyjcwt) 3

[Reviews / Approvals / Commitment](#_3dy6vkm) 3

[Definitions and Acronyms](#_1t3h5sf) 3

[Related documents / Referenced Materials](#_4d34og8) 4

[**SCOPE AND LIMITATIONS**](#_2s8eyo1) **4**

[Scope](#_17dp8vu) 4

[Limitation and Exclusions](#_26in1rg) 4

[**TESTING APPROACH**](#_lnxbz9) **5**

[Testing Objectives](#_35nkun2) 5

[Testing Types](#_1ksv4uv) 5

[**TEST ENVIRONMENT STRATEGY**](#_44sinio) **5**

[**TEST DATA STRATEGY**](#_2jxsxqh) **6**

[**TESTING ORGANISATION**](#_z337ya) **6**

[Testing Success Criteria](#_3j2qqm3) 6

[Entry Criteria](#_1y810tw) 6

[Exit Criteria](#_4i7ojhp) 6

[Test Case Management](#_2xcytpi) 7

[Test Case Management](#_1ci93xb) 7

[Priority Definitions](#_3whwml4) 7

[Test Case Design Standardization](#_2bn6wsx) 7

[Defect Management](#_qsh70q) 7

[Defect Workflow](#_3as4poj) 7

[Severity Definitions](#_1pxezwc) 7

[Priority Definitions](#_49x2ik5) 7

[Other Defect attributes](#_2p2csry) 8

[Progress Status and Reporting](#_147n2zr) 9

[Testing Requirements Traceability](#_3o7alnk) 9

[Tools](#_23ckvvd) 9

[Test Metrics and Analysis](#_ihv636) 9

[**ROLES AND RESPONSIBILITIES**](#_32hioqz) **10**

[Test Team Structure](#_1hmsyys) 10

[Staffing and Training Needs](#_41mghml) 10

[**SCHEDULE**](#_2grqrue) **10**

[High-level Test Schedule](#_vx1227) 10

[Schedule Dependencies](#_3fwokq0) 11

[**PROJECT CONDITIONS**](#_1v1yuxt) **11**

[Assumptions](#_4f1mdlm) 11

[Constraints and Dependencies](#_2u6wntf) 11

[**ISSUES AND RISKS**](#_19c6y18) **12**

[Issues](#_3tbugp1) 12

[Schedule Dependencies](#_28h4qwu) 12

[**TEST SUMMARY**](#_37m2jsg) **12**

# INTRODUCTION

## Purpose

The purpose of this document is to outline the high-level test strategy for the <PROJECT NAME> project, defining the preliminary test scope, high-level test activities, and organization, together with test management for the project. This test strategy is a planning tool that will provide the starting point for detailed test planning.

## Document Location

| **Location** |
| --- |
| <Please provide URL> |

## Revision History

| **Version** | **Date** | **Changed by** | **Summary of Changes** |
| --- | --- | --- | --- |
| <x.x> | <mm/dd/yyyy> | <Name Surname> | <xxx section removed/ added/ updated according to yyy reason> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## Reviews / Approvals / Commitment

| **Version** | **Issue Date** | **Reviewed by** | **Position** | **Review/Approval Date** |
| --- | --- | --- | --- | --- |
| <x.x> | <mm/dd/yyyy> | <Name Surname> | <Project role> | <mm/dd/yyyy> |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

## Definitions and Acronyms

| **Abbreviation** | **Definition** |
| --- | --- |
|  | *Project-specific or testing-related terms* |
|  |  |
|  |  |

## Related documents / Referenced Materials

| **#** | **Name** | **Location** |
| --- | --- | --- |
| 1. | <Document Name> | <URL> |
| 2. | *Specifications, test plan, etc.* |  |
| 3. |  |  |

# SCOPE AND LIMITATIONS

## Scope

*The scope of the testing must describe all aspects of the project deliverables that are in scope for the test effort.*

* *Business processes,*
* *Technology components,*
* *Training materials and learning aids,*
* *User documentation and so on*

<text here>

## Limitation and Exclusions

*List the areas of business and systems, processes and activities that have been excluded from the testing effort.*

<text here>

# TESTING APPROACH

*Using the solution development methodology and lifecycle selected earlier determine the approach to testing. The test approach describes the types of tests performed and the sequence of tests as executed in the test lab.*

<text here>

## Testing Objectives

The objectives for testing are:

*List the high-level test objectives that are major items to be proven as an outcome of the testing:*

* *Validate and verify system functionality works to specified requirements.*
* *Provide confidence for business owners that the solution meets business needs, etc.*
* *Establish a process to improve product quality.*

## Testing Types

* *Unit tests validate the smallest components of the system, ensuring they handle known input and outputs correctly. Unit test individual classes in your application to verify they work under expected, boundary, and negative cases.*
* *Integration tests exercise an entire subsystem and ensure that a set of components play nicely together.*
* *Functional tests verify end-to-end scenarios that your users will engage in.*
* *Other.*

*Specify Owner, Test object, Entry criteria, Exit criteria, Environment, Implementation approach, Tools/Techniques, etc.*

# TEST ENVIRONMENT STRATEGY

*This section focuses on the processes and procedures for requesting, provisioning and managing test environments.*

*List environment requirements and proposed techniques to cover them.*

*Define processes for requesting / scheduling builds.*

<text here>

# TEST DATA STRATEGY

This section describes the strategy for the provisioning and management of test data if there’s *need for such.*

*Categorize test data, define test data sources and mapping, Provisioning and Backup Process*

<text here>

# TESTING ORGANISATION

*This section documents the proposed process and governance of key processes.*

<text here>

## Testing Success Criteria

## Entry Criteria

Entry Criteria The following conditions are preferred before any testing can begin:

* Business requirement documentation is signed off by business owner
* Segments of development code are unit tested before released to the test environment
* Test artefacts are up to date and prioritized according to risk based analysis
* Test environment must be ready and accessible
* Actions for outstanding issues are finalized and assigned to the correct area
* Technical resources have been allocated and are available for support
* <add more>

## Exit Criteria

These are the minimum acceptable conditions before promoting the SBR solution to a live production environment:

* There are no outstanding high priority issues
* All Test Cases with a high and medium priority have been successfully executed
* Test coverage of % has been completed
* Approval to proceed to production environment received from
* During the course of the testing, certain tests (and by extension certain pieces of functionality) may not be tested due to the project's constraints. If tests cannot be executed, approval will be sought from the Test Manager and Project Manager(s) and this will be recorded, as well as the reasons why, in the appropriate test report
* <text here>

## Test Case Management

### Test Case Management

### Priority Definitions

### Test Case Design Standardization

It can include test case structure, attributes, naming convention, terminology and formatting accepted. Test Case Design Standard can exist as separate document, in this case refer to it in Test Strategy document.

## Defect Management

### Defect Workflow

### Severity Definitions

**Severity:** The degree of impact that a defect has on development or operation of a component or system.

| **Severity** | **Definition** |
| --- | --- |
| Critical | The defect that results in the termination of the complete system or one or more components of the system and causes extensive corruption of the data. The failed function is unusable and there is no acceptable alternative method to achieve the required results. |
| Major | The defect that results in the termination of the complete system or one or more components of the system and causes extensive corruption of the data. The failed function is unusable but there exists an acceptable alternative method to achieve the required results. |
| Moderate | The defect that does not result in the termination, but causes the system to produce incorrect, incomplete or inconsistent results. |
| Minor | The defect that does not result in the termination and does not damage the usability of the system and the desired results can be easily obtained by working around. |
| Cosmetic | The defect is related to the enhancement of the system where the changes are related to the look and feel of the application. |

### Priority Definitions

**Priority**: The level of (business) importance assigned to an item, e.g. defect.

At a high level, priority is determined by considering the following:

* Business need for fixing the defect
* Severity/Impact
* Probability/Visibility
* Available Resources (Developers to fix and Testers to verify the fixes)
* Available Time (Time for fixing, verifying the fixes and performing regression tests after the verification of the fixes)

| **Priority** | **Definition** |
| --- | --- |
| Immediate | Must be fixed immediately (work started in 1 hour) because of:  - Data Loss or Data Corruption  - Blocking business workflow  - Needed for time critical deadline (Hot Fixes, sale, expo, trade show)  - Numerous customer complaints about the issue  - Blocks further development or\and testing, no workaround |
| High | Must be fixed within current iteration because of:  - critical area of the system affected  - major feature(High priority Test Cases) broken or unavailable  - Numerous or\and major UI issues  - interrupts business workflow with workarounds (for all end users) |
| Medium | Must be fixed within current iteration because of:  - not a critical areas of the system  - minor feature(Medium priority Test Cases) broken or unavailable  - Numerous or\and minor UI issues  - interrupts business workflow with workarounds for some end users (not for all) |
| Low | Must be fixed within next iteration because of:  - trivial, cosmetic  - may affect business workflow |
| Very Low | The defect repair can be put off indefinitely. Should be put in to Product backlog for future. |

### Other Defect attributes

| **Attribute** | **Description** |
| --- | --- |
| Issue Type | *Defect, Sub-defect, In Dev Defect; NWAD; Security, etc. – determine what for should be used each issue type* |
| Defect Type | Type of defect from the list of available:  Functional: defects that should be created if functionality behavior deviates from required;  UI/Usability: defects that should be created if application UI deviates from mockups, for usability issues;  Performance: defects that should be created if performance of application deviates from required. |
| Environment | *Test configuration on which defect was found: device, web browser and its, URLs of applications, version of application, DB, etc.* |
| Description | Detailed defect description should contain:  Preconditions: set of actions and states that are already done/exist before steps that cause incorrect behavior.  Steps: Step-by-step instructions on how to reproduce defects.  Actual result: incorrect behavior of application as a result of steps. If the issue reproduces not always this should be mentioned here as well.  Expected result: correct behavior of application as a result of steps.  If there are attachments that represent the issue then the sentence “See attached <file name> file (or “See attached screenshot”) should be added to this item (in most cases for actual or expected result).  If there is any other information that relates to functionality in which a defect was found, explains correct behaviour in more details and helps to fix the issue faster it should be added (in most cases for expected result). |
| Fix Version/s | *Definition, who and when populate the field* |

### Progress Status and Reporting

*When and how, what information and to what extent is collected and provided such as: what test cases are executed, how long it took, how many test cases passed, how many failed, and how many are not executable.*

## Testing Requirements Traceability

*Add Traceability Matrix or define an approach on how to trace tests/incidents back to requirements.*

## Tools

*The processes and procedures for testing tools (including automation) is the accountability of both the Test Manager and their Leads to determine whether appropriate to introduce into the test execution effort.*

<text here>

## Test Metrics and Analysis

*Metrics need to be established for each testing phase. The metrics and measurement process must be agreed by the necessary stakeholders and will form the foundation for progress reporting.*

<text here>

# ROLES AND RESPONSIBILITIES

*You can add a Name column or provide a link to a document where listed team members with assigned project roles.*

| **Role** | **Responsibility / Accountability** |
| --- | --- |
| QC Lead |  |
| Manual QC Engineer |  |
| Tech Lead |  |
| Project Manager |  |
| Business Analyst |  |

## Test Team Structure

<text here>

## Staffing and Training Needs

*Add knowledge and experience desirable for each area.*

<text here>

# SCHEDULE

## High-level Test Schedule

<text here>

| **Test Activity / Milestone** | **Start Date** | **End Date** |
| --- | --- | --- |
|  | <mm/dd/yyyy> | <mm/dd/yyyy> |
|  |  |  |

## Schedule Dependencies

*Document any external dependencies that schedule relies upon*

<text here>

# PROJECT CONDITIONS

## Assumptions

*State any assumption made in developing the strategy.*

The following assumptions have been made in preparing this test strategy:

* Test resources will be provided in-house
* The test environment will be available at the start of the testing period
* Technical resources will be available to provide support for the resolution of project issues/defects
* All business requirements will be finalized, documented and incorporated into the documents which form the test base prior to the start of testing
* <text here>

## Constraints and Dependencies

*State any constraints and dependencies of the test strategy.*

The ability for the test team to deliver appropriate test coverage is contingent upon the following factors:

* There are no significant changes in priorities that require redeployment of resources
* The project scope remains relatively stable
* The test team is notified promptly of any changes in project scope so that impacts on testing can be properly assessed
* Development work is progressed and delivered within timeframes that enable full and detailed testing to take place
* The test team must be notified as soon as possible of any delays, potential or actual, in the project implementation so that resource allocation can be assessed
* <text here>

# ISSUES AND RISKS

## Issues

<text here>

| **ID** | **Issue Description** | **Resolution Plan** |
| --- | --- | --- |
|  |  |  |
|  |  |  |

## Schedule Dependencies

*List the key risks relating to the test strategy*

<text here>

| **ID** | **Risk Description** | **Rating** | **Resolution Plan** |
| --- | --- | --- | --- |
|  |  | High |  |
|  |  | Medium |  |
|  |  | Low |  |

# TEST SUMMARY

*This section must address what kind of test summary reports will be produced for the senior management along with the frequency.*

<text here>