

## Unit Testing Process

<b>1. UNIT TESTING PROCESS .....</b>	<b>2</b>
<b>2. PURPOSE AND SCOPE.....</b>	<b>2</b>
<b>3. ENTRY CRITERIA .....</b>	<b>2</b>
<b>4. EXIT CRITERIA.....</b>	<b>2</b>
<b>5. DELIVERABLES FROM THIS PROCESS:.....</b>	<b>2</b>
<b>6. USERS/CUSTOMERS OF THIS DOCUMENT:.....</b>	<b>2</b>
<b>7. PROCESS:.....</b>	<b>2</b>
7.1 PROCESS DETAILS: .....	2
7.1.1 <i>Methods of Execution of this Process:</i> .....	2
7.1.2 <i>Analyze detailed design:</i> .....	3
7.1.3 <i>Analyze Work Breakdown structure. (WBS)</i> .....	3
7.1.4 <i>Analyze coding standards:</i> .....	3
7.1.5 <i>Code Review:</i> .....	4
7.1.6 <i>Amendments / changes to specifications and subsequently the Programs:</i> .....	4
7.1.7 <i>Checking the Conformance to specification</i> .....	4
7.1.8 <i>White box test coverage:</i> .....	4
<b>8.0 APPENDIX “A1”: CODE REVIEW CHECKLIST.....</b>	<b>4</b>
<b>9.0 APPENDIX “A2”: UNIT TEST CASE REVIEW CHECKLIST .....</b>	<b>5</b>
<b>10.0. “APPENDIX “A3”: CHECKLIST FOR UNIT TESTING.....</b>	<b>6</b>
<b>11.0 “APPENDIX “A4”: UNIT TEST PROCEDURES .....</b>	<b>7</b>
<b>12 “APPENDIX “A5”: UNIT TEST PLAN .....</b>	<b>12</b>

W3SOFTTECH INDIA PRIVATE LIMITED

Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
Hyderabad, Telangana, India 500032.

PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : info@w3softech.com

UTP

## 1. Unit testing Process

## 2. Purpose and Scope

The purpose of this phase is to build a complete, quality software system from the “Blue-Print” provided in the design document i.e., the low level design document. This process shall define the manner in which the design modules are transformed into code and the coding standards that shall be adhered to. The purpose of this process is to facilitate the developers during coding phase.

## 3. Entry Criteria

This process will start once the following criteria are met/satisfied:

Software requirement specifications/BDDs are approved

Detailed design review is done; approval obtained for the same, and checked into Configuration Management (CM)

Work breakdown structure has been created/identified

High-level test plans are approved coding standard(s) are identified/selected

## 4. Exit Criteria

Release of compiled, linked & Unit tested code to CM

Code review document with review comments taken care in the code.

Unit test results approved.

## 5. Deliverables from this process:

Following are the deliverables from this process:

Unit Tested and reviewed and compiled code as per the specification.

Code review documents.

Unit test results.

## 6. Users/Customers of this document:

The customers / users of this document are:

Software development and engineering teams

Software Quality Assurance (SQA)

Review team

## 7. Process:

### 7.1 Process details:

#### 7.1.1 Methods of Execution of this Process:

Each Developer studies the allocated Design specification, coding standards before the actual construction (Refer: QMS Coding Standard Processes for C, C++, and JAVA).

W3SOFTECH INDIA PRIVATE LIMITED

Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
Hyderabad, Telangana, India 500032.

PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : [info@w3softech.com](mailto:info@w3softech.com)

UTP

Confidential

The Developer writes the code in line with the coding standards adopted for the project

The code should be compiled, linked & made error free

The code should be then released for Code Inspection. The code review checklist is provided in Appendix "A1"

After Code inspection, the code is given for thorough review and verification as per the Unit design specifications

Unit testing should be done as per the associated Test cases. The test results shall be recorded. Unit test checklist is provided in Appendix "A2"

Metrics such as the total no. of lines in code, deviation in the effort, schedule should be collected. The defects found during Code walk-through, unit testing should also be collected for Causal Analysis

After incorporating, the review comments and regenerating the code, release the same for the Module/Integration testing phase

#### **7.1..2 Analyze detailed design:**

Coding process shall start with the first activity of analyzing reviewed and approved detailed design specification from the System Architect. Effective methods for requirement analysis are used to identify and derive the software requirements. The result of requirement analysis and the rationale for the selected alternative are documented.

Design is analyzed to ensure they are feasible and appropriate to implement in software, clearly stated, consistent with each other, testable, and complete (when considered as a set). Rationale behind the selection of any method will be indicated if there is more than one methodology.

Problems encountered in coding are identified and reviewed with the group responsible for the detailed design, appropriate changes are made to the software requirements and High level design document.

#### **7.1.3 Analyze Work Breakdown structure. (WBS)**

Work breakdown structure is identified during the design phase. Here the entire scope of software is broken into smaller self-managed units, which can be coded by respective developers.

#### **7.1.4 Analyze coding standards:**

Coding standards are available as part of QMS to facilitate developers, in maintaining uniformity across development teams. This uniformity is in terms of naming conventions, commenting styles etc. This helps in understanding the software for enhancements/modifications.

W3SOFTECH INDIA PRIVATE LIMITED

Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
Hyderabad, Telangana, India 500032.

PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : info@w3softech.com

UTP

### 7.1.5 Code Review:

Coding shall be reviewed as per coding standards. Planning for Code walkthrough / coding review is done at project management plan level. Review comments are mentioned for attending by respective developer. The review team shall be constituted by the project Manager.

### 7.1.6 Amendments / changes to specifications and subsequently the Programs:

The change request form, which is raised for making changes to specs, is analyzed for impact analysis. The trigger for making changes in Design, Project management plan, and test plans, coding etc is sent to respective team members. Here developer receives input for code changes either directly from the change request form or from the designer who is incorporating the changes in design.

### 7.1.7 Checking the Conformance to specification

When the coding is complete from the developer end, the same is tested for correctness. The Software requirement specification and the design becomes the reference for testing the coding / program. During the actual design of the system, the test cases are written. These test cases are executed one or more times and the results of each are captured. Defect arising out of these cycles are then logged into defect tracking system.

### 7.1.8 White box test coverage:

- Statement coverage
- Branch coverage
- Condition coverage
- Every define/use path for every variable attribute

Using White box testing methods,

- Test cases can be derived that all independent paths within a module have been exercised at least once.
- Exercise all logical decisions on their *true* and *false* sides.
- Executes all loops at their boundaries.
- Exercise internal data structures to assure their validity.

## 8.0 Appendix “A1”: Code Review Checklist

S. No.	Activities
--------	------------

W3SOFTECH INDIA PRIVATE LIMITED  
Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
Hyderabad, Telangana, India 500032.  
PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : info@w3softech.com

UTP

Confidential

1.	Has the specification properly been translated into code?
2.	Is every part of the code consistent with the unit specification?
3.	Has proper use of language conventions been made?
4.	Is code compliant with coding standards for language style, comments, and module prologue?
5.	Are structured programming constructs used throughout?
6.	Are there multiple entry points?
7.	Are all logic paths correct?
8.	Are all comments unambiguous and correct?
9.	Are all data types and data declarations proper?
10.	Are all physical constants correct?
11.	Are local data structures properly defined and commented?
12.	Is the external interface correctly coded?
13.	Is the interface consistent with the architectural design?
14.	Is the logical complexity reasonable?
15.	Has error handling been specified?
16.	Are any error messages unclear to the user?
17.	Are all input arguments used?
18.	Are all output arguments produced?
19.	Are all local variables used only after initialization?

## 9.0 Appendix “A2”: Unit Test Case Review Checklist

S. No.	Activities
1.	Does each requirement that applies to the routine have its own test case?
2.	Does each design element that applies to the routine have its own test case?
3.	Has each line of code been tested with at least one test case? Has this been verified by computing the minimum number of tests necessary to exercise each line of code?
4.	Have all Defined-Used data-flow paths been tested with at least one test case?
5.	Has the code been checked for data-flow patterns that are unlikely to be correct, such as Defined-Defined, Defined-Exited, and Defined-Killed?
6.	Has a list of common errors been used to write test cases to detect errors that have been common in the past?
7.	Have all simple boundaries been tested--maximum, minimum, and off-by-one boundaries?
8.	Have compound boundaries been tested, that is, combinations of input data that

W3SOFTECH INDIA PRIVATE LIMITED  
Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
Hyderabad, Telangana, India 500032.  
PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : info@w3softech.com

UTP

Confidential

	might result in a computed variable that is too small or too large?
9.	Do test cases check for the wrong kind of data?
10.	Are representative, middle-of-the-road values tested?
11.	Is the minimum normal configuration tested?
12.	Is the maximum normal configuration tested?
13.	Do the test cases make hand-checks easy?
14.	Is compatibility with old data tested?

### 10.0. “Appendix “A3””: Checklist for Unit Testing

S. No.	Activities
1	Unit tests case with appropriate data has been developed to exercise and validate all specified application requirements, functions, and objectives.
2	Has each line of code been tested with at least one test case? Has this been verified by computing the minimum number of tests necessary to exercise each line of code?
3	Have all Defined-Used data-flow paths been tested with at least one test case?
4	Has a list of common errors been used to write test cases to detect errors?
5	Have all boundaries been tested-maximum, minimum, and off-by-one boundaries?
6	Do test cases check for the wrong kind of data, for example, a negative number of employees in a payroll program?
7	Retesting is required to verify that the problem solution eliminates the problem and does not introduce new errors.
8	For each type of test conducted, the test results are compared with the expected results and recorded.

Reviewed By:	
Date:	

W3SOFTECH INDIA PRIVATE LIMITED  
 Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
 Hyderabad, Telangana, India 500032.  
 PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : info@w3softech.com

## 11.0 “Appendix “A4””: Unit Test Procedures

### 1. Scope

- To define the procedure for activities related to Unit Testing that involves verification and validation of deliverables in all products/projects
- Main objective of testing is to find errors. A successful testing is one that reveals as many as undiscovered errors to make the Software more rugged and reliable.

### Entry Criteria/ Inputs

- Unit Test Plan (UTP)
- Source Code to be tested
- Necessary Drivers, Stubs and Test Scripts (in case of Automatic Testing)
- Detailed Design Document, New Features list or impacted Features list (in case of Incremental development)

### Activity Details

- Based on the project needs and criticality of the module, Design Lead and Developer(s) should decide to develop an independent Unit Test Plan for each Unit/ Module under development.
- Unit Test Plan preparation involves:
  - Establishing a Test approach based on the nature of System
  - Developing Test Cases based on the Test approach
  - Identify the required Stop Criteria (Test Completion Criteria) for Testing
- Unit Test Plan should capture the requirement traceability information from the input documents like ERD and Detailed Design Document.
- Unit Test Plan should be prepared, reviewed and approved prior to commencement of Unit Testing.
  - As a best practice, the Unit Test Plan can be reviewed along with the Detailed Design Document and also during the Code Review.
- The approved Unit Test Plan should be placed under Configuration Management.

W3SOFTECH INDIA PRIVATE LIMITED

Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
Hyderabad, Telangana, India 500032.

PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : info@w3softech.com

## UTP

- Test Cases developed for Unit Testing should have a Test Case ID, Test Case Description and the expected behavior/ results.
- Drivers, Stubs, Test Scripts/ Code (for Automation) can also be reviewed (Peer Review).
- A README file for Unit Testing should be prepared, where applicable and provide the details to:
  - Compile the Unit/ Module
  - Execute the Test Cases
  - Interpret the results
- The Unit Test Plan describes the features and functionalities that are to be tested during Unit Testing. The typical tests that will be carried out during Unit Test include:
  - Functionality tests
  - Input validation
  - Message handling
  - Output validation
  - Error handling
  - Code coverage, if applicable
  - Branch conditions
- Develop Unit Test Cases to test the Functional and Structural aspects of the Unit under test.
- Identify Unit Test Cases to test the Basic Path.
  - Complexity of the Unit/ Module under test will give the total number of linearly independent paths in that Unit/ Module.
- Apply Equivalence Partitioning, Boundary Value Analysis and Error Guessing (Intuitive Testing) techniques to identify all the functional test cases.
  - Equivalence Partitioning
    - Identify the classes within a logic
    - Select Test Cases from each of the classes
    - Executing one Test Case is equivalent to executing any other Test Case in the same group
  - Boundary Value Analysis
    - Identify Test Cases to test the 'Edge' conditions of boundaries

W3SOFTECH INDIA PRIVATE LIMITED

Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
Hyderabad, Telangana, India 500032.

PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : info@w3softech.com



## UTP

- Helps to exercise the boundary values
- Complements Equivalence Partitioning technique
- Error Guessing
  - Apply intuition and experience to identify the test cases. (For instance, an input value of zero may reveal a "division by zero" error or one may want to try certain inputs based on knowledge of the design or implementation of the software.)
- The Unit Test Cases should be developed based on the Unit Test Plan, Detailed Design Document and Source Code.
- Unit Test Plan and Unit Test Cases should be prepared by the Developer (Coder) and verified by the Design Lead.
- Indicate the planned Code Coverage (Statement and Branch) in Unit Test Plan.

## Unit Test Execution

- Unit Testing should be carried out based on the approved Unit Test Plan document.
- Developer should conduct Unit Testing.
- The results of the test should be compared with the expected results and the differences should be resolved.
- Developer should analyze all the reported Unit Test Defects to identify the problems. The identified problems should be fixed and status of the defect should be appropriately changed before the Software is submitted for next round of testing. Problems that are not fixed should be documented in Unit Test Report and communicated to the Design Lead/ Manager.
- Automate the Unit Test Cases, which are necessary to repeat the sequence of actions many times.
- Developer should be responsible for maintaining an up to date Unit Test Plan and Unit Test Report.

## Recommendations

- Ensure independency of Unit Test Case development to validate implementation against requirements of the Unit/ Module. This can be done by allocating the task of Test Case development to Developers who are not involved in the Detailed Design and Implementation of that module.
- While developing Unit Test Plan, ensure clear identification of

W3SOFTECH INDIA PRIVATE LIMITED  
Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
Hyderabad, Telangana, India 500032.  
PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : info@w3softech.com

## UTP

- Item to be tested, which
  - Could be a specific program
  - A class of similar programs
  - An interface or a functionality/ business transaction
- Functionality/ Feature ID to be tested should
  - Outline all the aspects, which will be checked during the testing of the specific item or a group of items
  - Give Id (Identifier) for each feature to be tested so that it is referred in the Unit Test Cases
- Functionality/ Feature ID not to be tested should
  - Outline all the aspects, which will not be checked during the testing of the specific item or a group of items
- For Units that cannot be tested in isolation, focus on creating small test programs during Unit Test Plan preparation.
- Plan for the development of Stubs and/ or Drivers during preparation of Unit Test Plan, wherever necessary.
- List down the use of Simulators, Any external test data generators, Record and playback tools (for GUI testing), Coverage tools, Profilers, as applicable in Unit Test Plan.
- Recommended to achieve a minimum of 85% Code Coverage during Unit Testing.
- One of the effective ways to test a Code is to test the bottom-most function first. This should then be integrated with the calling function and test the calling functions. Bottom-up Test assures that the lower level modules are tested before testing the higher-level modules which invoke them.
- For multi-user systems, features such as security measures, data integrity and recovery measures must be thoroughly tested.
- One important thing to be kept in mind is that the testing strategy should be "destructive testing", rather than constructive since the main objective is to find as many defects as possible.
- Unit Test Report should be placed under Configuration Management for future references.
- After completing Unit Testing of all the modules within a Design Team, these modules can be integrated and a Multi-module Testing can be performed.

W3SOFTECH INDIA PRIVATE LIMITED

Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
Hyderabad, Telangana, India 500032.

PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : info@w3softech.com

## UTP

- Judge based on the Time and Effort savings before doing the Test Case Automation.
- Characteristics of a Good Tester:
  - Enjoys breaking things and get job satisfaction from finding problems/ bugs
  - Be creative and identify new and different ways of testing
  - Does not compromise
  - Won't give up and always find more bugs
  - Must be a good team player and work with other teams/ team members

## Work Items/ Outputs

- Unit Test Plan
- Unit Test Defect Log and Test Report
- Unit Tested Source Code

UTP

## 12 “Appendix “A5””: Unit Test Plan

### **Purpose of the document**

The purpose of this document is to explain the test cases which would be generated for testing different modules / program units of the application.

### **Description of the system**

This section should give a brief explanation about the system, its purpose, what it does and how is it going to be benefited to the client / user. It should also explain in brief the list of modules / sub systems which are under the scope of testing.

### **Module / Sub system name and description**

This section should give a detail explanation about the module / sub system, its purpose and functionality. How does it integrate with the main system / application, what kind of data is used as input and what kind of data processing is done within the module and what is the output from this module / sub system should be explained.

### **Program Unit / Screen name and description**

This section should give a detail explanation about the program unit / screen, its purpose and functionality. List the main data elements which would be required to run / execute this program unit / screen. This should explain how this program unit / screen maps to which section of the requirement in SRS.

W3SOFTECH INDIA PRIVATE LIMITED

Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
Hyderabad, Telangana, India 500032.

PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : info@w3softech.com

## UTP

### Flow path

### Entry criteria

This section should explain what are the data inputs required to enter / run / execute this program unit / screen. What program unit / screen calls this screen.

### Accessibility / User levels

This section should explain what levels of user can get access to this particular program unit / screen.

### Process / Validation / Verification

This section should explain what are the basic validation / verification required to be performed to the program unit / screen.

### Exit criteria

This section should provide details of how this program unit / screen should exit. It should also explain what is the correct way of exiting the program and ways of abruptly terminating this unit. This should also explain the risks of ending this unit abruptly like system crash, data loss, if it can affect the functioning of units, which this unit calls. It should mention if the unit ends in correct manner then what program unit / screen should be called.

### Test cases

This section should explain in detail what data elements will be tested, what is the test data required, what is the valid data, what data should fail the program and how should the program behave with range of data. All this should be documented and recorded in the table format provided below.

W3SOFTECH INDIA PRIVATE LIMITED

Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
Hyderabad, Telangana, India 500032.

PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : info@w3softech.com

UTP

W3SOFTECH INDIA PRIVATE LIMITED  
Suite No: 7C, Cynosure Tech Park, Near Flyover, Gachibowli, Serilingampally,  
Hyderabad, Telangana, India 500032.  
PH:040-29802628/29, Website : [www.w3softech.com](http://www.w3softech.com), Email : info@w3softech.com