

Test Management and Control

Sultan Aljahdali, Syed Naimatullah Hussain
College of Computers and Information Technology,
Taif University,
Taif, Saudi Arabia
aljadali@tu.edu.sa, s.naimatullah@tu.edu.sa

Nisar Hundewale, Azeemsha Thacham Poyil
College of Computers and Information Technology,
Taif University,
Taif, Saudi Arabia
nisar@computer.org, azeemsha.tp@gmail.com

Abstract— This paper evaluates a system or system component, by manual or automated means to verify that the system satisfies specified set of requirements. The Test Management and Control involves three major activities; test planning, test execution and defect management which needs to be planned and monitored in a structured manner to ensure delivery of quality product adhering to all project timelines. The purpose of this document is to present the methods, procedures, and approach to be used in the verification and validation for Test Management & Control. It consists of important activities like defining all the testing methodologies in detail for the test planning and lay down a process and procedure so that test plan will reflect all the activities which are going to get executed during the execution phase. The Test execution phase is the actual verification phase where you need to monitor the test case execution closely, and make sure that you are well within the defined Upper Control Limit (UCL) and Lower Control Limit (LCL). During the test execution phase which is clearly planned during the test planning phase, any deviation from these UCL & LCL requires a revised plan so that the deliverables are not going to jeopardy. So once we have clearly defined all the rules of the game during the planning phase and work on the same during the execution phase, we can certainly make sure that the product delivered will of high quality product. Managing Defects is also one of the very important phases all along the testing cycle. We need to pay extra attention to the defects getting fixed during the last few days of the testing cycle, so as to maintain the stability of the product. Do not fix the minor defects at the end of the testing cycle, unless they are critical and severe.

Keywords-component: Upper Control Limit (UCL), Lower Control Limit (LCL), Problem Report Document (PRD)

I. INTRODUCTION

Testing is the process of exercising or evaluating a system or system component, to verify that the system satisfies specified requirements. Testing is not intended to complete missing requirements or to improve design or fix code. Testing must be completely independent from these activities to ensure an objective assessment of the implemented solution. In this context, testing should span the entire project life cycle [5], and should be integrated into all phases of the project. The different phases of testing are discussed in detail in following section. Testing involves several activities like test design, test planning, test execution, problem reporting, re-testing etc. which is common to all the phases and which needs to be planned and monitored in a structured manner to ensure delivery of quality product adhering to all project timelines. Following Process

Flow Diagram displays the different steps of the Test management Process (Figure 1.).



Figure 1. Test Management Process

A. Test Design

This is the process step wherein the complete testing task need to be designed as to how testing will be carried out in each module and phase [1], [2]. Following steps can be used as a broad guideline in carving out the details for test condition to be considered:

1) Identification of Test Cycles:

Test Cycle can be defined as: “A grouping of Business Processes, with a high level of commonality, from a task (transaction) perspective, which will be used to manage project component and system testing efforts.

2) Sort Business Processes into Cycles

Each sub-team should go through an exercise of identifying scenarios, in business terms, which fall within the scope of a “Test Cycle”. Where required, existing Scenarios may be combined and new scenarios may be identified.

3) Identify / Document any overlooked Scenarios:

Test conditions; can be defined as a business situation, which may potentially arise when executing a transaction in Project. Each Scenario will have a comprehensive set of test conditions identified to ensure that the system’s functionality is configured to handle each of these business situations in an acceptable manner. Test Conditions will be reusable in multiple Tests Scenarios. (Note: Although Test Condition is reusable, a different test result may be expected each time the Test Condition is used in a different Test Scenario).

For example, one of the Scenarios can be Procure Standard Materials; several test conditions may be identified for this scenario. Sample Test Conditions, for this scenario may include:

- Procure standard materials.
- Procure raw materials with freight price condition.

- Procure standard material against refurbishment work order etc.

A test condition will consist of the following:

- Test plan reference id
- Test Conditions
- Expected result
- Actual result, to be filled during actual testing.
 - Review each scenario to ensure all scenarios are defined, in a logical order.
 - Identify tests conditions for each scenario.

B. Test Planning

1) Prepare Test Plan:

Test plan [5] is a tool to streamline entire test execution process. Test plan helps in organizing testing activities to accelerate the entire testing process in a more effective and efficient manner with the available technology. The test plan should contain test designs, test procedures, naming conventions, and other test attributes. The major objectives of a test plan are to:

- Create a set of testing tasks.
- Assign resources to each testing task.
- Estimate completion time for each testing task
- Document testing standards

A good Test Plan is one which:

- Is clear, complete, and specific & is developed and reviewed early.
- Specifies tangible deliverables that can be inspected.
- Sets realistic quality levels for testing objects & includes time for planning.
- Defines clear roles and responsibilities & recognizes learning curves.
- Synchronizes testing effort with dynamic requirements.
- Transforms design knowledge into testing foundation.
- Coordinates input from different sources & provides central reference point for all future test efforts.

Once all the test activities are identified, the same should be planned using a project management tool such as MS project, which will enable monitoring the test progress. Test plan document would be a deliverable to the client.

The following Figure 2. illustrates the process group's overlaps and variation within a phase:

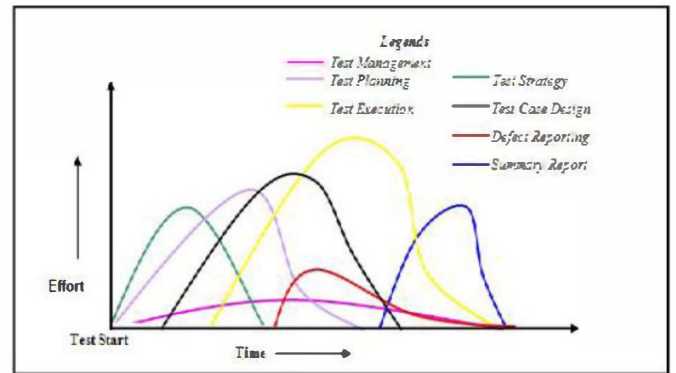


Figure 2. Overlap of Process Groups in a phase

2) Overlap of Process Groups in a Phase:

Test strategy [1], [2] is carried out in parallel with the requirement-gathering phase. Test planning would be carried out in parallel with the project planning activity. In parallel with design phase, Test case design and test data preparation would be carried based on appropriate proven methodologies. Defect reporting uses defect-tracking tool to log defects resulting from these tests.

C. Test Execution:

1) Executing Tests and Documenting Actual Test Results:

This is the step where actual validation is done to ensure that the configured system will support the client's business environment. This step should also ensure that the project handles all identified potential business conditions, which may arise, in an acceptable manner.

Once the entire Test Cycles, Test Scenarios and Test Conditions are defined, the next step is to execute those tests and documenting the actual result. The following steps should be the guidelines to carry out this activity -

- Review the test scenarios and conditions to be tested.
- Review test data and modify or add data for each test condition to be tested, if required.
- Review expected test results and modify if required
- Enter data required for successful execution of test condition & execute test condition.
- Document actual test results & document the defects, if any (100% defects documentation is must) and follow defect resolution process (as required).

The test steps to be followed should typically be provided by business process scripts created in the design stage. For certain test conditions, special test steps may be spelled out.

Upon completion of testing activities, actual results should be documented for all test conditions. For all conditions tested successfully, this can be the last step in the testing process. For all conditions, which are not successfully tested, defect resolution process steps must be followed, before the testing cycle is complete. All test results which have a status of "Failed Test", will have a corresponding Problem Report

Document (PRD), and will have to go through the Defect Resolution Process.

2) *Monitoring & Control of Test Execution (Quality Assessment)*

Upper Control Limits (UCL) and Lower Control Limits (LCL) [4] are depicted in Figure 3. The value of charting test case execution is to provide key quality assessment information in a manner that quickly determines whether the product quality is within expected bounds. This is accomplished quantitatively by identifying the key metrics to be tracked and the boundaries used for control.

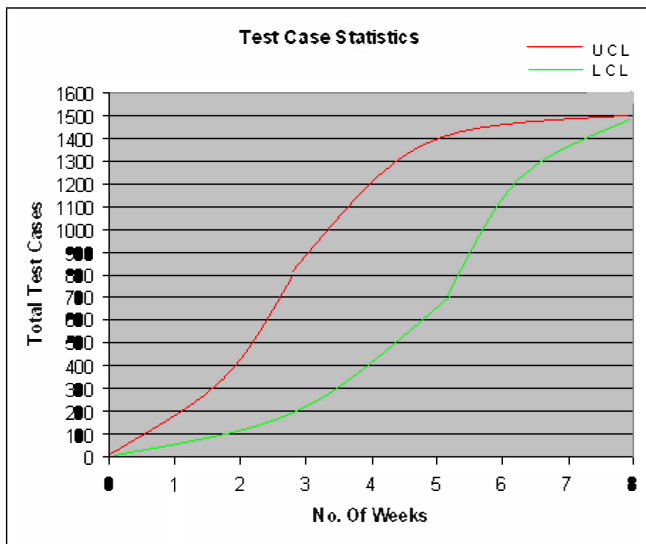


Figure 3. Test Case Statistics representing UCL and LCL

The following table (TABLE I.) shows the UCL and LCL and planned test case execution for this test cycle.

Weekly Activities	Week 1	Week 2	Week 3	Week 4	Week 5 Soft Freeze	Week 6 Hard Freeze	Week 7	Week 8- PT Sign Off / SHIP
UCL	9%	23%	49%	59%	72%	88%	95%	100%
LCL	1%	7%	21%	32%	48%	72%	86%	98%
% Executed	10%	20%	40%	50%	75% Overall	100%	100%	100%

TABLE I. SYSTEM TEST WEEKLY BOUNDARIES

As shown in above graph & table Test Execution should be within the UCL & LCL. Any deviation from these UCL & LCL requires a revised plan so that the deliverables are not going to jeopardy.

3) *Test Audit :*

Results of the tests, which are critical to the business, should be audited to avoid a defect becoming a Fault. A fault is a problem, which passes the Verification, Validation or testing prescribed for it in the test environment but is reported in the

production environment. Faults impact users and compromise the business. Therefore it becomes very important to audit all the critical Test Cycles and the relevant documentation, wherever necessary. To identify critical Test Cycles, all the cycles can be grouped together using ABC classification according to the criticality of cycle to the business.

D. *Managing Defects:*

1) *Reporting & Tracking Defects:*

Defect is a deviation from specification or standard. Strictly speaking anything that causes customer dissatisfaction can be termed as a defect [3]. To ensure that testing in each phase of Project implementation is successful and the final product meets the user's needs, it is very important to review and validate all testing activities throughout the project life cycle so that the interim deliverables meets their input specification.

It is important to report each defect whether it is minor, major or cosmetic. It helps in improving the Test Conditions and capturing the missing scenarios for the future testing.

2) *Defects Types are:*

- Data Inconsistency
- Loss of data
- Loss of functionality
- Cosmetic
- Network
- Hardware Performance
- Incorrect output.

3) *Defect Severity Levels:*

- Severity 1 (Work Stopper, No workaround, Resolve immediately).
- Severity 2 (Work Stopper, Workaround, Give High attention).
- Severity 3 (Not work Stopper, Normal Queue).
- Severity 4 (Cosmetics, Low Priority).
- Severity 5 (Modification, New Requirement, Defer).

4) *Analyzing Defects:*

The goal of analysis is to determine the source of the error. The analysis is often done with respect to the functional/technical design of the original requirement. The root cause analysis technique should be employed for this purpose. And the findings of this analysis should be incorporated in the system/process for potential continuous improvement.

5) *Defect Resolution:*

One of the goals of testing is to identify and resolve any defect, which is identified with respect to the deviations in the expected system behavior. Defects will arise in testing where

actual results do not equal expected results. Each defect maybe resolved in a different manner.

CONCLUSIONS

One of the innovative techniques which is mentioned above for monitoring and control of the test execution using the UCL & LCL methodology has been very effective and helpful to track the status of the test execution process and plan the test planning effectively. Soft Freeze & Hard Freeze is the two important time zones to maintain the stability of the product do not fix the minor defects at the end of the testing cycle, unless they are critical and severe.

ACKNOWLEDGEMENT

The authors gratefully acknowledge the contributions of Mr. Arif Ansari, senior delivery manager, Wipro Technologies for his constant support and encouragement for successful completion of this research paper.

REFERENCES

- [1] William e. Perry, "Effective Methods For Software Testing", 2ND Edition
- [2] Rex Black , "Managing the Testing Process ",Wiley, 2nd Edition, 2002
- [3] Cem Kaner," Lessons Learned in Software Testing", Robert Ipsen, 2nd Edition, 2002
- [4] <http://www.webproforum.com/vertel/full.html>
- [5] Pankaj Jalote, "An Integrated Approach to Software Engineering", Springer 2005, 3rd edition