

# Software Testing Mentor

[www.softwaretestingmentor.com](http://www.softwaretestingmentor.com)

**ISTQB Foundation Level and Software Testing Training**

[www.softwaretestingmentor.com](http://www.softwaretestingmentor.com)

# **Module 5**

## **Test Management**

### **Session 2 – Test Planning and Estimation**

# Test Planning and Estimation

In this  
session  
we will  
learn  
about

- Test Planning
- Test Estimation
- Test Approaches or Strategies

# Purpose of Test Planning

Test plan is the project plan for testing work to be done in project

Test plan guides our thinking

Test plan forces us to confront the challenges that await us and focus our thinking on important topics

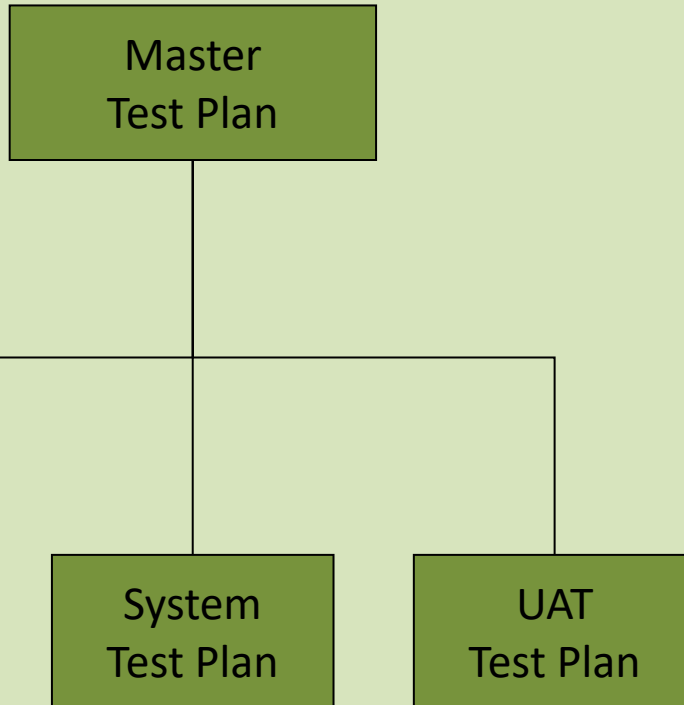
Test plan serves as the vehicle for communication with other team members and stakeholders

Test plan helps to manage change

# IEEE 829 Standard Test Plan Template

- Test-plan identifier
- Introduction
- Test items
- Features to be tested
- Features not to be tested
- Approach
- Item pass/fail criteria
- Suspension criteria and resumption requirements
- Test deliverables
- Testing tasks
- Environmental needs
- Responsibilities
- Staffing and training needs
- Schedule
- Risks and contingencies
- Approvals

# Test Planning Hierarchy



- Master Test Plan – High level plan for the project
- Phase Test Plan – Individual Test Plans for each phase of testing

# Activities of test planning

Test planning tasks that you need to carry out during test planning are as follows:

- Determine what is in scope and out of testing scope
- Determine the test objectives
- Determining project and product risks
- Constraints which may affect testing (Resources, budget, time etc.)
- Most critical things to consider for the product or project
- Overall testing Approach
- Integration and coordination of testing activities
- Assigning resources and Test Scheduling
- Test deliverables to be produced
- Defect logging, Change and Configuration Processes
- Determining entry and exit criteria

# Factors for determining entry/exit criteria

## Acquisition and supply

- Availability of resources (Testers, software, hardware)

## Test items

- The state in which test item should be before starting/ending test execution

## Defects/Bugs

- Total defects found/remaining/defect detection rate/defects resolved



# Factors for determining entry/exit criteria cont.

## Test Cases

- Total test cases executed, total passed/failed/blocked/skipped etc.

## Coverage

- Percentage of test basis covered by testing/code coverage etc.

## Quality

- Status of important quality characteristics of software

# Factors for determining entry/exit criteria cont.

## Money/Budget

- Cost of finding defects in current execution or after in production

## Risks

- Any undesirable outcomes – shipped with some untested features or loss in market share if shipped too late

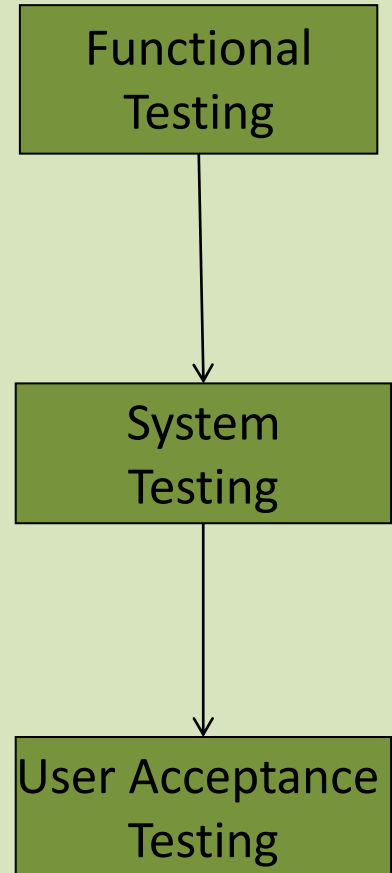
# Entry and Exit Criteria

The entry criteria helps you to decide when can you start the testing for a particular testing phase.

The exit criteria helps you decide when to stop the testing for a particular phase

The Exit Criteria from one stage forms part of the Entry Criteria into the next testing phase

Entry/exit criteria are the true Quality Checks of the actual software testing



# Entry Criteria

Typical  
Entry  
Criteria for  
a testing  
phase may  
consist of:

- Build verification test passed – Stable build
- Availability of required test data and test environment, resources etc.
- All required test documents completed and signed off

# Exit Criteria

Typical exit criteria for a test phase may include factors like:

- Test coverage has been attained
- No outstanding defects (Severity 1 & Severity 2)
- Cost and time (Time to market) constraints
- No critical residual risks

# Test Estimation

Testing is often a subproject within the large project

You should start with work breakdown structure that identifies stages, activities and tasks

First breakdown the testing project into phases using the fundamental test process

- Planning and control
- Analysis and design
- Implementation and execution
- Evaluating exit criteria and reporting
- Test closure

Identify activities within each phase and then identify tasks and subtasks

Now estimate how long these tasks and subtasks will take to finish

# Estimation Techniques

There are 2 estimation techniques covered in ISTQB foundation level

- Consultative approach – Taking inputs from people who will do task and domain experts
- Metrics Based - Analyzing metrics from the past

# Consultative approach

Individual contributors and experts prepare the work breakdown structure for the project

After that team works together to understand effort, duration, dependencies and resource requirement

This is a bottom up approach because you start with lowest of work breakdown structure - Task

This is mostly followed previous metrics about project are not available

Disadvantage is that you have to rely on estimation done by the task owner or an expert



# Metrics based approach

In this approach metric from similar previous projects are used for estimation

The simplest approach for this approach is to take developer – tester ratio as we had in similar past projects

More reliable approach is classify the project in terms of size and complexity and then compare how long similar project took in past

Average time required to execute 1 test case in past and estimating the total effort

Other sophisticated approaches can also be applied

- Build mathematical models to compare key parameters from past projects and then estimating effort for current tasks

# Factors affecting test effort

## Product

- Sufficient product documentation
- Complexity of the project
- Size of the project

## Process

- Availability of test tools
- Availability of test and development environment
- Process maturity in organization
- People factors : Team skills, relationships
- Time Pressures

## Test Results

- Defects found during test execution
- Defects failing re-testing
- Rework required due to changing requirements

# Test Approaches or Strategies

Choosing the right test strategy is very important for success of the test project

Major types  
of test  
strategies  
that are  
commonly  
used are:

- Analytical
- Model Based
- Methodical
- Process or Standard Compliant
- Dynamic
- Consultative or Directed
- Regression Averse

# Test Approaches or Strategies Cont.

## Analytical

- In analytical approach analysis of the risk or specification documents form the basis for test design
- Risk based test strategy
- Requirements based test strategy

## Model Based

- Test is based on some defined model for example mathematical model to upload data on servers
- If the system conforms the defined model then system is assumed to be working

# Test Approaches or Strategies Cont.

## Methodical

- Adhere to pre planned and systemized approach that have been defined in house based on prior experience of testing the application
- Uses checklists which suggests major areas for testing
- Might also follow industry standards

## Process or Standard Compliant

- Follow the standard for your testing for example IEEE 829 standard
- Rely on the externally developed standard approach of testing
- Can follow well defined standards such as V-Model or Agile development

# Test Approaches or Strategies Cont.

## Dynamic

- Uses lightweight set of testing guidelines which address weaknesses in software
- Focus is to find as many defects as you can in test cycle
- Exploratory testing technique is one technique used in Dynamic approach

## Consultative or Directed

- In this approach testing is based on the guidance from developers/technology experts and domain experts

# Test Approaches or Strategies Cont.

## Regression Averse

- Regression averse strategy have the automated set of regression tests which find any regression defects
- Tester tries to automate most of the tests of system functionality to ensure nothing has broken with any changes in software

# Selecting the Right Test Strategy

Some of the factors to consider while selecting Test Approach are:

## Risk

- For established software which is evolving slowly regression averse strategy is right approach
- For new software risk based approach is best fit

## Skills

- Skills available in test team
- Cannot pick regression averse if no automation skillset is present in team

## Objectives

- Testing must satisfy the needs of stakeholders
- If focus is to find more and more defects then dynamic strategy makes sense



# Selecting the Right Test Strategy Cont.

## Regulations

- If Regulatory requirements need to be met then methodical test strategy that satisfies the regulations is fit

## Product

- If good and extensive product documentation is available then requirements based analytical strategy is good fit

## Business

- Business consideration are also important in choosing test strategy
- If an existing system can be used to model new system then model based approach should be used

# Conclusion

In this session we will learn about

## Test Planning

- Purpose of Test Planning
- Activities of test planning

## Test Estimation

- Estimation Techniques: Metrics based and consultative approach

## Test Approaches or Strategies

**THANK YOU!!!**