



## Document History

Date	Course Version No.	Software Version No.	Developer / SME	Reviewer(s)	Approver	Change Record Remarks
	0.1D	NA				Content Creation
	0.1	NA				Review
May-2009		NA	Priya Rane			Material Revamp
June-2011	1.0	NA	Hema G.			Material Revamp
April-2014	1.1	NA	Dayanand Patil	Samir Attar Shilpa Bhosle		Material Revamp
June-2016	2.0	MS Office 2010	Neelima P. & Leena S Pangarkar	Leena S Pangarkar & Shilpa Bhosle	Shilpa Bhosle	Post-Integration Material Revamp

## Course Goals and Non Goals

### ■ Course Goals

- At the end of this program, participants gain an understanding of Verification & Validation process in project
- Participants get an understanding of different testing approaches, techniques & types
- They also learn how to create effective test cases using the different testing techniques to get a good test coverage of a software application
- Participants get an understanding of Importance of monitoring progress in testing process & different project metrics

### ■ Course Non Goals

- This course does not cover automation process of testing.



## Pre-requisites

- None

## Intended Audience

- Test Engineers, Software Engineers and Senior Software Engineers



### Day Wise Schedule

- Day 1
  - Lesson 1: Fundamentals of Testing
  - Lesson 2: Types of Testing Techniques & Test Case Design
- Day 2
  - Lesson 2: Types of Testing Techniques & Test Case Design (Cont.)
- Day 3
  - Lesson 2: Types of Testing Techniques & Test Case Design (Cont.)
- Day 4
  - Lesson 3: Testing throughout the Software Life Cycle
  - Lesson 4: Test Management & Test Case Execution
- Day 5
  - Lesson 5: Testing Metrics
  - Lesson 6: Tool Supporting for Testing

## Table of Contents

- Lesson 1: Fundamentals of Testing
  - 1.1 Some Facts
  - 1.2 Introduction to Software Testing
  - 1.3 Software Testing - Definitions
  - 1.4 Need of Software Testing
  - 1.5 Error-Failure-Defect
  - 1.6 Causes of Software Defects
  - 1.7 Cost of Software Defects
  - 1.8 What does Software Testing reveal
  - 1.9 Importance of Software Testing
  - 1.10 Importance of Testing Early in SDLC Phases
  - 1.11 Testing and Quality
  - 1.12 Quality Perceptions

## Table of Contents

- Lesson 1: Fundamentals of Testing
  - 1.13 Seven Testing Principles
  - 1.14 Economics of Testing
  - 1.15 How Testing is conducted?
  - 1.16 Software Testing – Then (Past)
  - 1.17 Software Testing – Now (Present)
  - 1.18 Scope of Software Testing
  - 1.19 Factors influencing the Scope of Testing
  - 1.20 Risk Based Testing
  - 1.21 Project Risks
  - 1.22 Product Risks
  - 1.23 Need of Independent Testing
  - 1.24 Activities in Fundamental Test Process



## Table of Contents

- Lesson 1: Fundamentals of Testing
  - 1.25 Attributes of a good Tester
  - 1.26 Psychology of Testing
  - 1.27 Code of Ethics for Tester
  - 1.28 FS SBU: Focus on Testing
  - 1.29 Testing Roles in iTEAMS
  - 1.30 Limitations of Software Testing

## Table of Contents

- Lesson 2: Types of Testing Techniques & Test Case Design
  - 2.1 Verification and Validation
  - 2.2 Types of Testing Techniques
  - 2.3 Static & Dynamic Testing Techniques
  - 2.4 Introduction to Static Testing Techniques
  - 2.5 Static Testing Techniques – Defects Detected & Benefits
  - 2.6 Review Process Success Criteria
  - 2.7 Introduction to Dynamic Testing
  - 2.8 Types of Dynamic Testing Techniques
  - 2.9 White Box Test Techniques
  - 2.10 Black Box Testing
  - 2.11 Static vs. Dynamic Testing
  - 2.12 A good Test Case

## Table of Contents

- Lesson 2: Types of Testing Techniques & Test Case Design
  - 2.13 Test Case Lifecycle
  - 2.14 Test Case Design Techniques
  - 2.15 What is test data?
  - 2.16 Properties of Good Test Data
  - 2.17 Test Data team
  - 2.18 Test data lifecycle
  - 2.19 Requirement and Planning
  - 2.20 Request Process
  - 2.21 Test Data Creation Techniques
  - 2.22 Test Data From Production Data
  - 2.23 Test Data Life Cycle - Maintenance
  - 2.24 Test Data in STLC - Staggered with test case Design

## Table of Contents

- Lesson 2: Types of Testing Techniques & Test Case Design
  - 2.25 Test data in STLC -Standalone phase between Test Case Design and Test Case Execution
  - 2.26 What is Positive Testing?
  - 2.27 Advantages/Limitations of positive testing
  - 2.28 What is negative testing?
  - 2.29 Advantages/Limitations of negative testing
  - 2.30 Positive & Negative test scenarios
  - 2.31 What is Basic test?
  - 2.32 Example on Basic test
  - 2.34 What is Alternate test?
  - 2.35 Example on Alternate test
  - 2.36 Importance of writing positive, negative, basic, alternate test while designing test cases
  - 2.37 Best practices for test case maintenance

## Table of Contents

- Lesson 3: Testing throughout the Software Life Cycle
  - 3.1 Testing throughout the Software Life Cycle
  - 3.2 Introduction of SDLC and V-Model
  - 3.3 SDLC and V-Model
  - 3.4 Iterative Life Cycles
  - 3.5 Rapid Application Development
  - 3.6 Rational Unified Process (RUP) Phases
  - 3.7 RUP Phases and Disciplines
  - 3.8 Agile Development – Extreme Programming (XP)
  - 3.9 Testing Phases
  - 3.10 Introduction of Component Testing
  - 3.11 Component /Unit Testing
  - 3.12 Introduction of Integration testing

## Table of Contents

- Lesson 3: Testing throughout the Software Life Cycle
  - 3.13 Why Integration Testing is Required?
  - 3.14 Types of Integration testing
  - 3.15 Top Down Integration Testing
  - 3.16 Top Down Integration Testing
  - 3.17 Bottom Up Integration Testing
  - 3.18 Top Down vs. Bottom Up Testing
  - 3.19 Introduction to System Testing
  - 3.20 Types of System Testing

## Table of Contents

- Lesson 4: Test Management & Test Case Execution
  - 4.1 Test Planning
  - 4.2 Test Plan Contents (IEEE 829)
  - 4.3 Test Planning Activities
  - 4.4 Entry Criteria for Functional Testing
  - 4.5 Test Case Execution - Pre-execution activities
  - 4.6 Types of Test Environment
  - 4.7 Before starting Execution
  - 4.8 Test Case Execution
  - 4.9 Exit Criteria for Functional Testing
  - 4.10 Test Estimation Techniques
  - 4.11 Factors affecting Test Effort
  - 4.12 Independent Testing
  - 4.13 Roles & Responsibilities - Working as Test Leader
  - 4.14 Roles & Responsibilities - Working as a Tester

## Table of Contents

- Lesson 5: Testing Metrics
  - 5.1 Monitoring the Progress
  - 5.2 Metrics of Test Progress
  - 5.3 Reporting Test Status
  - 5.4 Test Control
  - 5.5 Configuration Management & Configuration Control
  - 5.6 Products for Configuration Management in Testing
  - 5.7 Definition of Metrics
  - 5.8 Need of Metrics
  - 5.9 Metrics for Testing
  - 5.10 Types of Metrics
  - 5.11 Types of Metrics – Project Metrics
  - 5.12 Types of Metrics – Process Metrics
  - 5.13 Types of Metrics – Productivity Metrics
  - 5.14 Types of Metrics – Closure Metrics



## Table of Contents

- Lesson 6: Tool Supporting for Testing
  - 6.1 Tool support for Testing
  - 6.2 Test Tools Classification
  - 6.3 Tool Support for Management of Testing and Test
  - 6.4 Tool support for Static Testing
  - 6.5 Tool support for Test Specification
  - 6.6 Tool support for Test Execution & Logging
  - 6.7 Tool support for Performance & Monitoring
  - 6.8 Tool support for specific Testing Needs
  - 6.9 Need of Software Testing Tools
  - 6.10 Potential Benefits of using Tools
  - 6.11 Risks of using Tools
  - 6.12 Special Considerations for some Types of Tools
  - 6.13 Introducing a Tool into an Organization

## References

- Student material:
  - Class Book (presentation slides with notes)
  - Lab book
- Book:
  - Testing Computer Software – Cem Kaner
  - Software Testing in the Real World – Edward Kit
  - Effective methods for Software testing – William E. Perry
  - Software Engineering -A Practitioner's Approach – Roger S. Pressman
  - Software Testing Techniques – Boris Beizer
- Web-site:
  - <http://www.softwaretesting.org>
  - <http://www.onestoptesting.com/introduction/>



## Next Step Courses

- Automation testing

