

ISTQB Software Testing & Advanced Test Automation Engineering

Duration: 6 Days

Day 1 – Foundations and Software Lifecycle

Module 1: Fundamentals of Testing

- What is Testing?
 - Test Objectives
 - Difference between Testing and Debugging
- Why is Testing Necessary?
 - Contribution of Testing to Success
 - Role in Quality Assurance
 - Errors, Defects, Failures, and Root Causes
- Testing Principles
- Test Activities, Testware and Roles
 - Test Activities and Process
 - Testware and Traceability
 - Roles in Testing
- Essential Skills and Good Practices
 - Generic Skills Required for Testing
 - Whole Team Approach
 - Independence in Testing

Module 2: Testing Throughout the Software Development Lifecycle

- Testing in the SDLC Context
 - Software Development Lifecycle Models
 - DevOps and Testing
 - Shift Left Approach
 - Retrospectives and Process Improvement
- Test Levels and Types
 - Unit, Integration, System, Acceptance Testing
 - Functional vs Non-functional Testing
 - Confirmation Testing and Regression Testing
- Maintenance Testing

Module 3: Static Testing

- Static Testing Basics
 - Types of Work Products
 - Value of Static Testing
 - Static vs Dynamic Testing
- Feedback and Review Process
 - Stakeholder Feedback
 - Review Process Activities

- Roles and Responsibilities in Reviews
 - Review Types
 - Success Factors for Reviews
-

Day 2 – Test Design Techniques

Module 4: Test Analysis and Design

- Test Techniques Overview
 - Classification of Test Techniques
 - Black-Box Test Techniques
 - Equivalence Partitioning
 - Boundary Value Analysis
 - Decision Table Testing
 - State Transition Testing
 - White-Box Test Techniques
 - Statement Testing and Coverage
 - Branch Testing and Coverage
 - Value of White-box Testing
 - Experience-based Test Techniques
 - Error Guessing
 - Exploratory Testing
 - Checklist-Based Testing
 - Collaboration-based Test Approaches
 - Collaborative User Story Writing
 - Acceptance Criteria
 - Acceptance Test-Driven Development (ATDD)
-

Day 3 – Test Management and Tools

Module 5: Managing the Test Activities

- Test Planning
 - Purpose and Content of a Test Plan
 - Tester's Role in Iteration and Release Planning
 - Entry and Exit Criteria
 - Estimation Techniques
 - Test Case Prioritization
 - Test Pyramid
 - Testing Quadrants
- Risk Management
 - Risk Attributes
 - Project and Product Risks
 - Product Risk Analysis and Control
- Test Monitoring, Control and Completion
 - Metrics in Testing

- Test Reports: Purpose, Content, Audience
 - Communicating Test Status
- Configuration Management
- Defect Management

Module 6: Test Tools

- Tool Support for Testing
- Benefits and Risks of Test Automation

Wrap-up & Q/A

- Recap of Key Concepts
- Sample Exam Questions
- Final Clarifications

Day 4 – Foundations and Architecture of Test Automation

1. Introduction and Objectives for Test Automation

- Purpose of Test Automation
- Advantages and Disadvantages
- Automation Across SDLC Models
- Tool Selection for Different System Types

2. Preparing for Test Automation

- Infrastructure Configuration Needs
- Automation in Different Environments
- Evaluation and Selection of Automation Tools
- Technical Findings and System Analysis

3. Test Automation Architecture (Part 1)

- Capabilities of a Test Automation Architecture
- Designing Test Automation Solutions
- Layering of Test Automation Frameworks

Day 5 – Implementation, CI/CD, Reporting

4. Test Automation Architecture (Part 2)

- Approaches for Automating Test Cases
- Design Principles and Design Patterns in Automation

5. Implementing Test Automation

- Guidelines for Automation Development & Pilots
- Risk Analysis and Mitigation
- Maintainability Factors

6. Implementation and Deployment Strategies

- CI/CD Pipeline Integration
- Configuration Management for Testware
- API Infrastructure Dependencies

7. Test Automation Reporting and Metrics (Part 1)

- Data Collection Methods
 - Analyzing Automation Data
-

Day 6 – Verification and Continuous Improvement

8. Test Automation Reporting and Metrics (Part 2)

- Constructing and Publishing Test Progress Reports

9. Verifying the Test Automation Solution

- Verification of Tool Setup and Environment
- Expected Behavior and Troubleshooting
- Static Code Analysis for Quality

10. Continuous Improvement

- Identifying Improvement Opportunities
- Analyzing Deployed Automation Solutions
- Restructuring Testware to Align with SUT
- Strategic Use of Automation Tools