Test Planning and Design (3 Days)

For teams that need a practical, risk-based approach to planning and designing tests across agile and traditional lifecycles. Effective testing begins well before execution. Plan with purpose and design tests that matter.

This course helps participants create test strategies and plans that make sound use of time, people, environments, and test assets. It also develops the test design skills needed to cover requirements, business events, interfaces, and risk.

The course treats testware as a living body of work—test plans, cases, scripts, data, matrices, and supporting materials that must be created, maintained, controlled, and improved throughout the lifecycle. Concepts are reinforced through a continuing case study and hands-on exercises.

Who Should Attend

- Test managers and team leaders
- Test engineers
- Test designers and analysts
- Testers
- Developers
- End Users who contribute to quality and acceptance decisions

What Participants Will Learn

Plan strategically: Define objectives, scope, resources, environments, procedures, schedules, and success factors.

Design stronger tests: Apply equivalence partitioning, boundary value analysis, decision tables, state transitions, scenarios, and experience based techniques

Prioritize by risk: Assess product, project, and business risk and use it to guide test depth and execution order.

Build reusable test assets: Organize and maintain test cases, scripts, data, matrices, and regression suites across releases and sprints.

Business Value

- Get more value from the organization's existing investment in testing.
- Connect requirements, risk, test planning, test design, and execution.
- Improve test coverage while making priorities and tradeoffs explicit.
- Strengthen practical testing expertise through realistic, cumulative exercises.

Course Topics

Module 1 – Testing Foundations | 1 hours

Establish a shared vocabulary and a practical view of testing across the lifecycle.

- testing, QA, and QC
- defects and requirements
- test levels
- verification and validation
- static and dynamic testing
- V model
- black box, white box, and gray box testing

Module 2 – Agile Testing Methods | 1.5 hours

Connect testing work to agile delivery, sprint planning, feedback, and technical debt.

- Scrum and agile variants
- agile testing goals and characteristics
- testing quadrants and pyramid
- release and sprint planning
- acceptance criteria
- in sprint regression
- retrospectives

Module 3 – Developing Quality Requirements and User Stories | 1.5 hours

Improve the testability and clarity of requirements and stories before implementation.

- quality and testable requirements
- ambiguity and word meanings
- stakeholder involvement
- brainstorming and prototyping
- change control
- requirements reviews

Module 4 – Static Testing with Reviews | 1.5 hours

Use early stakeholder feedback and structured reviews to prevent and find defects sooner.

- static versus dynamic testing
- review objectives and value
- review types and roles
- success factors
- workflow walkthrough

Module 5 - Basic Test Planning | 1.5 hours

Translate testing principles into a usable strategy and clear objectives.

- basic testing process
- test strategy components
- critical success factors
- test objectives
- planning guidelines

Module 6 - The Test Planning Process | 2 hours

Build an adaptable test plan that reflects lifecycle, context, and risk.

- strategy and objectives
- resources and environments
- procedures, functions, and interfaces
- scripts, cases, and data
- test matrix
- schedule and plan completion

Module 7 - Risk Based Test Planning | 1.5 hours

Focus limited test effort on what matters most to the business and the product.

- project, product, and business risk
- risk ownership and timing
- risk based prioritization
- user story backlogs
- METS approach
- release and escalation decisions
- sprint execution mapping

Module 8 - Test Case Development | 5 hours

Design, document, and maintain test cases that cover functions, rules, states, and business journeys.

- test cases and scripts
- equivalence partitions
- boundary values
- decision tables
- state transitions
- error guessing and checklists
- business scenarios and transaction threads
- test cycles
- collaborative and story based design
- ongoing maintenance

Module 9 - Regression Testing | 1 hour

Create a risk informed regression approach that protects changed and integrated behavior.

- regression process and prerequisites
- coverage decisions
- integration points and workflows
- selection by release risk
- practical execution tips

Module 10 - Unit Testing | 1.5 hours

Understand preparation, functional design, and structural coverage at the unit level.

- when unit testing occurs
- preparation needs
- functional tests
- logic based structural tests
- coverage levels
- behavioral tests

Module 11 - System Testing | 2 hours

Bring the course together in an end to end system test plan.

- system objectives, functions, and critical requirements
- interfaces and integration points
- scripts, cases, schedules, and resources
- representative plan outline
- manual and automated approaches
- defect reporting

Practice throughout the course

Participants work through a continuing case study that includes backlog selection, ambiguity analysis, a workflow walkthrough, test strategy creation, system test objectives, risk-based execution order, multiple test design techniques, regression selection, logic testing, and an end-to-end plan for an online retail ordering system.

A complete view of thoughtful testing

The course follows the way strong testing decisions are actually made: understand the context, improve the inputs, plan the work, assess the risk, design meaningful coverage, and maintain what must be retested. Participants leave with techniques they can adapt to their own products, teams, and delivery methods.

Practical	Adaptable	Integrated
Exercises turn concepts into decisions and usable work products.	Guidance applies across agile, traditional, and hybrid lifecycles.	Planning, risk, requirements, design, regression, and test levels are treated as one system.