

END TO END SOFTWARE TESTING COURSE

End to End Software Testing is a 2-day course that presents an intensive, practical guide to requirements-based software testing throughout the development lifecycle. It will provide you with a toolkit of information and strategies to help improve your testing process.

What is the aim of the course?

- It has a strong focus on risk-based testing, with practical techniques for coping with the quality vs schedule conflict
- It reflects the most recent trends in software testing, incorporating anecdotal examples from experienced testers actively working in the field
- It stresses the benefits of a structured testing strategy that provides for test involvement from the very beginning of a project.

What will participants learn?

- How to plan an efficient testing strategy
- How to manage the testing process
- How to write tests, perform tests and manage test results
- Tips, traps, what works, what doesn't
- Practical ways of dealing with day-to-day testing tasks, though exercises and discussions.

To Register

- Enquire about availability:
 - Phone **(02) 9957 6577**; or
 - Email susan.parker@ivvaust.com.au

About the course

- \$1395 per person (GST included)
- 2 day workshop
- Small groups (max 12) for individual attention
- 8:45 to 5:00 each day
- Available as a public course or as an in-house course on your premises

Who can benefit from the course?

- Anyone responsible for planning, managing or conducting requirements-based software testing
- This includes software testers and test managers, project managers, developers and development managers
- It is suitable for all experience levels.

How is the course material presented?

- The material is presented through the use of lecture, hands-on exercises, and articles on a variety software testing issues
- The course notes contain a set of checklists and worksheets that testers can take away and use on their projects to help implement the tasks discussed during the course.

Name: _____

Position: _____

Organisation: _____

Address: _____

Phone: _____

Email address: _____

Project Manager: _____

Course date and location: _____

Notes: Registered attendees who do not attend or who cancel two weeks or less prior to the start date of the course are liable for the entire fee.

Collection of this information is subject to our privacy policy. For more information see our website.

End To End Software Testing Course Outline

Day 1

Overview of Software Testing

- What is software testing and why do it?
- Why test early and often?
- Why use a structured, well-planned approach?
- Human factors - peers, managers, customers

Test Strategies

- Strategic planning - defining the overall approach
- Writing test guidelines/ a master test plan
- Defining test activities
- Identifying and managing testing risks
- Budgeting, scheduling and estimating

Test Documentation

- Why write test documents? (objectives)
- Using the IV&V test document template
- Tailoring standards to suit your needs

Reviews

- Reviewing documents (using checklists)
- Reviewing tests - quality gates
- Reviewing progress

Requirements

- The role of testing in requirement reviews
- The “Reminder” Functional Specification
- Identifying good and bad (and ugly) requirements (an exercise)
- Requirement attributes
- Requirements tracing
- What if there are no written requirements?

Unit Testing

- Verifying that the code is implemented as designed
- Documenting unit tests
- The hand-over from Developer to Tester

Day 2

Requirements-based Testing

- Managing the software risks by prioritising

Functional Testing

- Verifying that the requirements are met
- Planning tests - a “Reminder” exercise
- Designing tests using a variety of test types and methods
- Writing test procedures for the “Reminder” application

System Testing

- Validating that the system is fit for purpose
- System testing re-used and COTS software

Test Tools

- Classes of tools
- When are tools a good idea and a bad idea?
- A sensible approach to automation

Running Tests

- Controlling the test environment
- Running tests and debugging procedures
- Problem reporting

Test Metrics - the feedback loop

- What should I collect? What can it tell me?
- How should I measure / graph the data?
- Analysing test results - when to stop testing

Regression Testing

- What is regression testing?
- What do you retest?

Final Acceptance

- Making sure you are ready for system release
- Final acceptance - audits and tests
- Writing test reports

References and Suggested Reading