

Testing Business Systems: From Inception to Acceptance

Duration: 2 Days

Overview

In today's world, your organization is only as good as your information systems. As a consequence, testing business information systems has become a corporate priority, not another assignment for an overworked information technology department. People with sufficient business acumen to decide what is important to the business community need to be involved in defining, designing, testing, and evaluating evolving information technology solutions.

This course presents topics, tools, and techniques on how to plan, structure and execute static and dynamic testing efficiently and effectively in the real world. Effective, efficient testing needs to be planned and budgeted at the beginning of your project and is based on a well-thought out strategy. A test plan encompasses a set of test scenarios and cases that are designed to reduce the business risk of deploying the technology to the end-users. Test steps are documented in test scripts. Test strategies, test plans, test cases and test scripts are components of your testing methodology. You need a consistent approach for developing each component individually and for maintaining it as an integral part of your test environment.

Approach

This seminar is based on two simple philosophies:

- a) the technology components of your business systems should be tested at the appropriate level of detail to reduce your exposure; and
- b) testing doesn't follow development, it is an integral part of the creative process.

The course applies adult learning theories of experiential and discovery learning, meaning you learn by experience and sharing of ideas with peers and knowledgeable instructors. From the selection of a testing strategy based on business risks to acceptance of the solution by the business community, the methods, concepts, techniques, tools, and tricks of the testing trade are presented, experienced and discussed.

Audience

The target audience includes subject matter experts, quality analysts, test engineers, test managers, quality engineers, project managers, end-users responsible for acceptance testing, developers, and anyone who is interested in assuring that an evolving business system is built with the appropriate level of quality to avoid costly production problems.

Developed and presented by:

Hathaway & Associates, Inc.
16057 Tampa Palms Blvd. W., # 197
Tampa, FL 33647

"Effective Business Use of Information and Systems"

Telephone: (813) 973-3046
Fax: (813) 864-0131
Email: training@thehathaway.com
Websites: www.thehathaway.com
www.businessanalysisbooks.com

Outline

BTI The Testing Challenge

Duration ≈ 1 hour

Is testing the developer's job? Developers cannot test the business premise behind the system, they cannot test the business use of information technology, nor are they knowledgeable about the business risks associated with software failure. Is testing the subject matter expert's job? The SME does not understand the intricacies of information technology, how to locate bugs, or what to do with one once it is found. Assessing the true value of the information system in the context of the business world is a full-time job that requires qualified and motivated people who want to prove that the technological solution works well enough to be worthy of your organization's trust.

1. What is a business system and why do you care?
2. When should business system testing start?
3. How do you define the major testing activities and deliverables?

BTS Selecting a Solid Testing Strategy

Duration ≈ 2 hours

Testing is not something that happens after the system is developed and before the end-users see it – it is an on-going sequence of activities that start the day the project is initiated and end when the solution is in production. This requires an assessment of business risks, the underlying technology, organizational structure, the development process, and the people who are impacted by the solution.

1. What do you have to consider in selecting a testing strategy?
2. Do web-based and/or client/server technologies influence the strategy?
3. Does your organizational culture play a role in how you test?

BTP Proper Test Plans Prevent Poor Performance

Duration ≈ 2 hours

Test planning flushes out the details of how to implement your selected testing strategy. From developer-based unit testing to end-user acceptance, your testing techniques have to adapt to fit the evolving state of the solution. The nature of testing at each stage has to be properly understood to make the most of whatever opportunities are presented for identifying and eliminating potential problems.

1. When do you need performance, usability, and configuration testing?
2. What does a good test plan encompass
3. Who does unit, integration, system, and acceptance testing and when?

BST Static Testing: Defect Avoidance versus Defect Detection

Duration ≈ 2 hours

Instead of waiting until the developers make the errors to start detecting them, imagine you could eliminate them before they happened! Static testing covers every aspect of assessing the decision-making process that defines the solution under development before software is ever produced. Walk-throughs, inspections and reviews are all different dimensions of this powerful, money-saving, testing concept.

1. What is the difference between a walk-through, an inspection, and a review?
2. How can you increase the number of defect detected by static testing?
3. Who is responsible for implementing and conducting static testing?
4. What types of errors are common and how can you find them?

BTE Test Engineering: Knowing Where You're Going

Duration ≈ 2 hours

No two tests are created equal; some find more errors than others. Test engineering describes tests, documents the setup criteria, scripts the execution steps, and establishes how to quickly evaluate whether the solution passed the test or failed. The best tests are those that will find critical errors in the least amount of time.

1. How can you develop tests before the software is finished?
2. Which tests are more likely to identify errors and why?
3. How can you test if you have no requirements?
4. When do you need to create a test script?

BTX Test Execution: Testing made Real

Duration ≈ 2 hours

The test team initiates test execution when they receive their first working component and finishes it when the goals defined in your test strategy have been achieved. Test execution is an iterative process that restarts after any change to the evolving solution. Properly executed testing minimizes the number of iterations needed to create a working solution worthy of being implemented and deployed to the business community.

1. What is the difference between a test, a test case, a test scenario, and a test condition?
2. You've found a defect; now what?
3. How do you classify and categorize bugs to decrease your testing effort?

BAT After the Testing: Lessons Learned

Duration ≈ 1 hour

Testing, like most processes, is best learned by experience. Unfortunately, all too often, we neglect to document significant dimensions of the experience and are forced to learn them again and again. Intelligent tracking of the testing effort, bugs detected and corrective actions taken will augment the learning process and avoid making the same mistakes next time.

1. How do testing statistics help you improve your testing methodology?
2. How can you change your organization's approach to testing?
3. What kind of automated tools can expedite the testing effort?

Appendices

- A. Case Study
- B. Index
- C. Bibliography
- D. Evaluation and Name Tent

Objectives

Attend this section:

To be able to:

BTI	The Testing Challenge	Understand the role of developers and SME's in testing List the 7 major steps of a well-defined testing methodology Recognize unit, integration, system, and acceptance testing
BTS	Selecting a Solid Testing Strategy	Determine the critical business factors that define the strategy Select the appropriate type of coverage to reduce business risks Defend the need for software testing based on business risks Use requirements management principles to focus the test effort Assess technology impact, i.e., web-based, client/server, mainframe
BTP	Proper Test Plans Prevent Poor Performance	Define the strategic and scheduling components of a test plan Optimize your testing resource usage Create a test plan that is synchronized with your strategy Assemble the 17 key elements of effective test plans
BST	Static Testing: Defect Avoidance versus Defect Detection	Reduce the cost of software development through early detection Determine when to use walk-throughs, reviews, and inspections Select the appropriate approach for evaluating critical deliverables Evaluate business requirements for testability
BTE	Test Engineering: Knowing Where You're Going	Mix and match sources of set-up test data for maximal coverage Apply equivalence classes, boundary values and probable errors Create audience-focused test scripts to maximize repeatability Identify the minimal set of cases to achieve defined testing goals
BTX	Test Execution: Testing Made Real	Create an effective problem report to aid defect removal Use defect categorization to reduce test execution time Capture critical test execution information consistently Organize tests for efficient execution
BAT	After the Testing: Lessons Learned	Generate testing statistics that make a difference Define 7 activities to improve your approach to testing Evaluate the potential effect of automating testing activities Define the role of a test engineer in a JAD/JAR session