Build-Your-Own Testing Training Week

Pair two courses in the same location and save up to $300!

See page 4 for details.

Accelerate Your Career & Empower Your Team

SOFTWARE TESTING TRAINING

SATISFACTION Guaranteed

- Relevant, Up-to-Date Content
- Best Practices
- Small Classroom Workshop Environment
- World-Class Expert Instructors

NEW FALL 2008 SCHEDULE

TEST ENGINEERING

- Lean-Agile Testing Practices
- Systematic Software Testing
- Mastering Test Design
- Software Testing Certification
- Just-in-Time Software Testing
- Performance, Load, and Stress Testing

TEST MANAGEMENT

- Test Management
- Test Process Improvement
- Visual Studio® Team System Training
- ... and Many More Courses Inside

SQE TRAINING
www.sqetraining.com

Public Training
On-site Training
eLearning
Consulting

R.E.P. Project Management Institute

ISTQB

CHOOSE THE BEST LEARNING OPTION FOR YOU AND YOUR TEAM

Public Training Courses

We provide the widest selection of specialized training courses—and we're expanding our selection all the time. Developed by top industry consultants, all courses are based on the latest industry practices and updated regularly to reflect current technologies, trends, and issues. Industry experts teach all SQE Training courses. We enhance your experience with expert instruction, content tailored to students' needs, and group discussions. We offer you the best training value in the software industry.

On-site Training

Looking for ways to save training and travel dollars? Take advantage of the cost-effective convenience of on-site training. To provide your team the training they need without sacrificing project schedules or incurring travel time and expenses, bring our training to your facility.

eLearning

The perfect solution for software professionals with travel and time constraints, eLearning offers classroom value with the convenience of self-paced instruction on the Web.

Free Seminar

Attend a free seminar on Risk-Based Testing at Training Week locations. Learn a few principles of modern testing including the importance of using risk analysis to prioritize tests and to formulate contingency plans.

Certification Training

Attend internationally recognized certification training presented by industry experts. Our accredited training courses help prepare you for ISTQB™ certifications.

Training Week

Maximize the impact of your training by combining courses in the same location to create a customized training week. Look for this TW marked on Training Week courses. Pair two and save up to $300!

For more information about SQE Training's courses and management consulting services, please visit www.sqetraining.com.

On-site Training

Looking for ways to save training and travel dollars? Take advantage of the cost-effective convenience of on-site training to get your team the training they need without requiring them to sacrifice project schedules or incur travel time and expense. Our on-site training offers many benefits:

- Save time and money—Bring team training to your location.
- Manageable workloads—Schedule training around your projects, not the other way around.
- Customizable content—Offer your team a training curriculum that adheres to your corporate goals, technology environment, and business needs.
- Consulting services—Learn from instructors who are world-class consultants with exceptional qualifications and a broad range of real-world experience. Augment your training programs with SQE Training's consulting services.
- Small groups—Benefit from focused training that offers your team members individual attention with plenty of time for questions. Class sizes can range from 6–25 people.
- Employee development—Develop the talent already on your team, increase employee satisfaction—and save company dollars.

If you have six or more people to train, consider the advantages of on-site instruction.

For additional information call 904.278.0524 or email onsitetraining@sqe.com.

WHO'S BEHIND THE TRAINING?

SQE Training provides the widest selection of specialized software training courses available. Developed and taught by top industry consultants, all courses are based on the latest industry practices and updated regularly to reflect current technologies, trends, and issues. Find the training you need for software testing, development, management, requirements, and security. **www.sqetraining.com**

Easy to Register

Online:
www.sqetraining.com/register

Phone:
888.268.8770 / 904.278.0524

Email:
sqeinfo@sqe.com

Ways to Save

Take advantage of the different "Ways to Save" on training using our discount programs listed below. Purchase valuable software quality training for your whole team and save.

ON-SITE ADVANTAGE

Bring any course to your location for team training. Take advantage of this cost-effective convenience for your team of six or more. For a free quote, contact us at onsitetraining@sqe.com or 888.268.8770 or 904.278.0524. See page 6 for more details.

BUILD-YOUR-OWN TRAINING WEEK

Attend a week's worth of specialized training in the same location and save up to $300. See page 4 for details.

POWERPASS

Add a StickyMinds.com PowerPass to any training purchase for only $125. With a PowerPass you save $100 on all future training registrations. What's a PowerPass? Visit www.StickyMinds.com/PowerPass to learn more.

*For **Group Discounts** or more details on our discount policy, contact the SQE Training Client Support Group at sqeinfo@sqe.com or call 888.268.8770 or 904.278.0524.*

TRAINING CURRICULUM

P PUBLIC | O ON-SITE | eLEARNING | F FREE SEMINAR | C CERTIFICATION | TW TRAINING WEEK COURSE

Learn the latest skills and techniques through SQE Training's courses delivered in a high-powered workshop setting. Plan your training curriculum and improve your whole team. Various learning options allow you to take each course in the method that works best for you.

TEST ENGINEERING

TEST MANAGEMENT

VISUAL STUDIO® TEAM SYSTEM TRAINING

PMI Project Management Institute R.E.P.

SQE Training has been reviewed and approved as a Registered Education Provider by the Project Management Institute.

What Our Students are Saying ...

"The best applied knowledge course I've ever taken. I had a wonderful learning experience; the instructor was fun and lively; and the course allowed me to be certified. What more can I say but FANTASTIC."

Matthew O'Rourke, Sr. Project Manager
MPA Associates
On "Software Testing Certification"

"Clearly, we were very pleased with this course and felt lucky to have gotten the guy who literally 'wrote the book' on the subject. Kudos to Rick Craig. His unique combination of experience, expertise and personality really helped turn this potentially dry subject matter into a relevant and engaging training experience. We hope to get him back again for more training. Thanks SQE."

Ron Graham, Director of U.S. Dept. of Health and Human Services
On "Systematic Software Testing"

"This course provided me with a solid understanding of the concepts of testing as well as when and how to apply them. Its explanation of orthogonal arrays and McCabe complexity matrix was especially helpful. I recommend this course for new and veteran testers. The sections on requirements, system testing, and basis paths would be very helpful for developers as well."

Becki Bloch, Test Engineer
Alliance Data Systems
On "eMastering Test Design"

"This class and instructor have been the only one who has succeeded in: showing the material with interest; keeping the group interested; not reading his PowerPoint presentation word for word. Dale Perry seemed very knowledgeable in this field and material. Thank you for doing such a great job!"

Rita Webb, Software Quality Analyst
Catalina Marketing Corporation
On "Software Testing Certification"

BUILD-YOUR-OWN TESTING TRAINING WEEK

FALL 2008 SCHEDULE

TESTING

September 8–12, 2008
New York/New Jersey Area

September 15–19, 2008
Washington, DC

October 20–24, 2008
San Francisco, CA

November 17–21, 2008
Tampa, FL

MICROSOFT® VISUAL STUDIO® TEAM SYSTEM TRAINING

September 22–25, 2008
Chicago, IL

October 13–16, 2008
Washington, DC

BUILD-YOUR-OWN TRAINING WEEK

Maximize the impact of your training by combining courses in the same location to create a customized training week.

Pair any of these courses in the same location to build a week of training and save up to $300.

TESTING

MONDAY	TUESDAY	WEDNESDAY	THURSDAY	FRIDAY
Systematic Software Testing			Mastering Test Design	
Software Testing Certification—Foundation Level			Lean-Agile Testing Practices	
Writing Testable Requirements			Software Security Testing and Quality Assurance	
Just-in-Time Software Testing			Requirements-Based Testing	
Test Management			Performance, Load, and Stress Testing	
			Exploratory Testing in Practice	
			Test Process Improvement	

MICROSOFT VISUAL STUDIO® TEAM SYSTEM TRAINING

MONDAY	TUESDAY	WEDNESDAY	THURSDAY
Visual Studio® 2008 Team Foundation Server		Testing Applications with Visual Studio® Team System 2008	

LEARNING PATHS

Are you looking to build your current skill set or become more specialized? Create a training program to fit your career path. For more information on any of these courses, visit www.sqetraining.com or contact us at 888.268.8770 or 904.278.0524.

	FUNDAMENTALS	SPECIALIZED
Test Engineer	**Systematic Software Testing** *3 days* **Software Testing Certification—Foundation Level** *3 days* **ISTQB™ Certified Tester** *Foundation Level* **Mastering Test Design** *2 days* (Also available in eLearning format)	**Lean-Agile Testing Practices** *2 days* **Just-in-Time Software Testing** *3 days* **Performance, Load, and Stress Testing** *2 days* **Writing Testable Requirements** *3 days* **Requirements-Based Testing** *2 days* **Software Security Testing and Quality Assurance** *2 days* **Exploratory Testing in Practice** *2 days* **Testing Applications with Visual Studio® Team System 2008** *2 days*
Test Manager	**Test Management** *3 days*	**Test Process Improvement** *1 day* **Lean-Agile Testing Practices** *2 days* **Testing Applications with Visual Studio® Team System 2008** *2 days* **Visual Studio® 2008 Team Foundation Server** *2 days*

TRAINING LOCATIONS

We're bringing our training to you! For questions regarding course registration and schedules, please call our Client Support Group at 888.268.8770 or 904.278.0524.

ARIZONA

December 2–4, 2008	Phoenix	Software Testing Certification

CALIFORNIA

September 28–30, 2008	Anaheim	Software Testing Certification
October 21–23, 2008	Sacramento	Software Testing Certification
October 20–22, 2008	San Francisco	Software Testing Certification
October 20–22, 2008	San Francisco	Systematic Software Testing
October 20–22, 2008	San Francisco	Writing Testable Requirements
October 20–22, 2008	San Francisco	Just-in-Time Software Testing
October 20–22, 2008	San Francisco	Test Management
October 23–24, 2008	San Francisco	Mastering Test Design
October 23–24, 2008	San Francisco	Lean-Agile Testing Practices
October 23–24, 2008	San Francisco	Software Security Testing and Quality Assurance
October 23–24, 2008	San Francisco	Requirements-Based Testing
October 23–24, 2008	San Francisco	Performance, Load, and Stress Testing
October 23–24, 2008	San Francisco	Exploratory Testing in Practice
October 23, 2008	San Francisco	Free Seminar—Risk-Based Testing
November 18–20, 2008	Sunnyvale	Software Testing Certification

FLORIDA

October 7–9, 2008	Jacksonville	Software Testing Certification
November 4–6, 2008	Ft. Lauderdale	Software Testing Certification
November 17–19, 2008	Tampa	Software Testing Certification
November 17–19, 2008	Tampa	Systematic Software Testing
November 17–19, 2008	Tampa	Writing Testable Requirements
November 17–19, 2008	Tampa	Just-in-Time Software Testing
November 17–19, 2008	Tampa	Test Management
November 20–21, 2008	Tampa	Mastering Test Design
November 20–21, 2008	Tampa	Lean-Agile Testing Practices
November 20–21, 2008	Tampa	Requirements-Based Testing
November 20–21, 2008	Tampa	Performance, Load, and Stress Testing
November 20–21, 2008	Tampa	Exploratory Testing in Practice
November 20, 2008	Tampa	Test Process Improvement
November 21, 2008	Tampa	Free Seminar—Risk-Based Testing

GEORGIA

September 23–25, 2008	Atlanta	Software Testing Certification

ILLINOIS

September 22–23, 2008	Chicago	Visual Studio® 2008 Team Foundation Server
September 24–25, 2008	Chicago	Testing Applications with Visual Studio® Team System 2008

INDIANA

Sept. 30–Oct. 2, 2008	Indianapolis	Software Testing Certification

MARYLAND

November 4–6, 2008	Bethesda	Software Testing Certification

MASSACHUSETTS

August 26–28, 2008	Boston	Software Testing Certification

MINNESOTA

September 9–11, 2008	Minneapolis	Software Testing Certification

MISSOURI

October 14–16, 2008	Kansas City	Software Testing Certification

NEBRASKA

October 28–30, 2008	Omaha	Software Testing Certification

NEW YORK

September 8–10, 2008	New York/NJ Area	Software Testing Certification
September 8–10, 2008	New York/NJ Area	Systematic Software Testing
September 11–12, 2008	New York/NJ Area	Lean-Agile Testing Practices
September 11–12, 2008	New York/NJ Area	Mastering Test Design
October 14–16, 2008	Rochester	Software Testing Certification

NORTH CAROLINA

October 28–30, 2008	Charlotte	Software Testing Certification

OHIO

October 28–30, 2008	Cincinnati	Software Testing Certification

ONTARIO

October 7–9, 2008	Toronto	Software Testing Certification

PENNSYLVANIA

September 23–25, 2008	Philadelphia	Software Testing Certification
October 21–23, 2008	Pittsburgh	Software Testing Certification

UTAH

September 9–11, 2008	Salt Lake City	Software Testing Certification

WASHINGTON, DC

September 15–17, 2008	Washington	Software Testing Certification
September 15–17, 2008	Washington	Systematic Software Testing
September 15–17, 2008	Washington	Writing Testable Requirements
September 15–17, 2008	Washington	Just-in-Time Software Testing
September 15–17, 2008	Washington	Test Management
September 18–19, 2008	Washington	Mastering Test Design
September 18–19, 2008	Washington	Lean-Agile Testing Practices
September 18–19, 2008	Washington	Software Security Testing and Quality Assurance
September 18–19, 2008	Washington	Requirements-Based Testing
September 18–19, 2008	Washington	Performance, Load, and Stress Testing
September 18–19, 2008	Washington	Exploratory Testing in Practice
September 18, 2008	Washington	Free Seminar—Risk-Based Testing
October 13–14, 2008	Washington	Visual Studio® 2008 Team Foundation Server
October 15–16, 2008	Washington	Testing Applications with Visual Studio® Team System 2008

FREE SEMINAR!

RISK-BASED TESTING FOR SOFTWARE MANAGERS

In the best of circumstances, it is impossible to comprehensively test a software product. When you add competitive schedules and tight budgets, software managers are faced with the daunting task of trying to decide what and how much to test. Often the software manager's pleas for more time and/or resources fall on deaf ears, because the software manager cannot adequately measure the effectiveness of the testing effort. While there are no easy answers to these problems, there are strategies that managers can use to address these issues.

In this short session, Rick Craig introduces a few of the principles of modern testing including the importance of using risk analysis to prioritize tests and to formulate contingency plans. He also demonstrates how to improve on normal requirements-based coverage models and explains a couple of useful metrics to measure test effectiveness.

Seminar Hours
9 a.m. – 11 a.m.
Breakfast Included

F

Sept. 18, 2008	Washington, DC
Oct. 23, 2008	San Francisco, CA
Nov. 21, 2008	Tampa, FL

ON-SITE TRAINING

Looking for ways to save training and travel dollars? Take advantage of the cost-effective convenience of on-site training in a team-building atmosphere. If you have six or more people to train, consider the advantages of scheduling courses at your location. Employing an interactive workshop format, our world-class consultants/instructors are able to address many of your organization's specific issues. In addition, we can customize the course delivery to meet your unique situation. All of the SQE Training courses listed in this brochure can be taught at your location. A complete list of on-site offerings is provided on pages 6 and 7. For more information on any course, please visit **www.sqetraining.com/OnSite**

ON-SITE ADVANTAGE

For additional information or to receive a FREE quote, call 888.268.8770/ 904.278.0524, or email onsitetraining@sqe.com.

Testing

Software Testing Certification
Certified Tester—Foundation Level Training

Systematic Software Testing
A Risk-Based Approach for Producing Better Software

Mastering Test Design
Techniques for Developing Focused Test Cases

Creative Software Testing
Proven Testing Techniques for Fast-Paced Projects

Performance, Load, and Stress Testing
Issues and Solutions for Software Performance Testing

Implementing a Test Automation Framework
Learn to Select and Implement the Right Test Automation Framework for Your Needs

Requirements-Based Testing
A Disciplined Approach for Testing—and Improving—Software Requirements

Technical Reviews and Inspections
Find More Defects in Less Time: A Hands-on Course

Exploratory Testing in Practice
Plan, Design and Execute Tests Simultaneously to Find More Bugs—Faster

Lean-Agile Testing Practices
Rapid Delivery of High Quality Software

Software Security Testing and Quality Assurance
An Integrated, Risk-Based Approach to More Secure Software

Just-in-Time Software Testing
Powerful Tools for Fast-Changing Projects and Priorities

Writing Testable Requirements
Produce Better Requirements to Reduce Effort and Costs

Development

Scrum Master Implementation Workshop
Applying Lean-Agile Software Development Practices with Scrum

User Stories and Estimation in Agile Development
How to Write User Stories and Estimate Development Time

Design Patterns Explained
Principles, Practices, and Qualities of Good Design

Practical Test-Driven Development
A Revolutionary Approach to Software Design and Programming

Management

Managing Test Outsourcing
A Proven Approach for Assessment, Implementation, Management, and Monitoring

Test Management
What Every Test Manager Needs to Know

Managing the Test Process
Practical Tools and Techniques for Managing Software Testing

Leading Successful Software Projects
Essentials for Software Project Managers

Managing Software Risk
How to Safeguard Your Software Projects

Practicing Great Management
A Guide for Leading People and Projects to Success

Test Process Improvement
Practical Guidelines for Small and Large Test Organizations

Software Test Estimation
A Practical, Hands-on Course for Managers

Security

Software Security Fundamentals
Build and Deliver Bulletproof Software Applications

Defensive Programming—Java EE
Secure Software Development and Avoiding Common Mistakes

Defensive Programming—Core Java
Secure Software Development and Avoiding Common Mistakes

Architecture Risk Analysis
Threat and Ambiguity Analysis Lead to More Secure Software

Requirements

Essential Software Requirements
Techniques and Practices for Successful Projects

Mastering the Requirements Process
Ensure that You Build the System Your Customer Wants

Requirements Modeling
Use Models to Improve Your Requirements Gathering and Systems Analysis

Extending Requirements
Take Your Requirements to the Next Level

Introduction to CMMI® Measurement

Introduction to the Capability Maturity Model Integration® (CMMI®)
Practical Software Process Improvement with Staged and Continuous Approaches

Business-Driven Software Measurement
Develop and Improve your Software Measures and Metrics

Visual Studio® Team System Training

Testing Applications with Visual Studio® Team System 2008
Managing Lifecycle Testing

Visual Studio® 2008 Team Foundation Server
Understanding and Administering Microsoft® Team Foundation Server

SYSTEMATIC SOFTWARE TESTING

A Risk-Based Approach for Producing Better Software

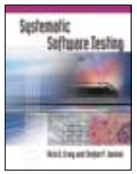
- Develop effective testing plans and strategies, execute them efficiently, and measure your results
- Design "testware" that finds important bugs more quickly and with less effort
- Prevent software defects and failures by integrating testing into your development process
- Improve your software testing practices and the quality of your organization's software
- Explore a flexible, risk-based approach to testing for both small and large organizations

PMI Project Management Institute R.E.P.

You will earn 22.5 contact hours or PDUs upon completion of this course.

The Best Fundamental Course for Any Test Professional

Learn the techniques necessary to develop and maintain a systematic, integrated software testing approach for your organization. This course details an adaptable and repeatable approach to testing that results in significantly improved software quality. Better planning, analysis, design, and implementation of tests result in happier clients and developers.



Take-Home Bonus

Each public course participant receives a copy of *Systematic Software Testing*. Order additional copies by visiting www.sqe.com/books.asp

Valid for public courses only.

The STEP™ (Software Test and Evaluation Process) approach described in this course emphasizes prevention of software defects and stresses continuous improvement for lasting benefits. The STEP process has been implemented in hundreds of testing organizations and integrated into many different software development lifecycle methodologies.

Know What You've Done—and When You're Done

Understanding and managing risk helps you focus on the important testing issues. Trace your tests back to requirements, design, and code to reveal what you have tested and what remains to be tested. This course leads you through test planning, test analysis, and test execution, showing you how to set—and then effectively satisfy—your testing goals.

Who Should Attend

The audience includes test professionals, test managers, project leaders, quality analysts, and developers. No specific testing prerequisites are assumed. However, attendees are expected to have some software experience.

3 Days Topical Outline:

Foundations of Modern Testing
Testing as a formal process
Testing within development lifecycles
Testing as a part of quality assurance
Importance of risk-based testing

STEP™—The Software Test and Evaluation Process
STEP architectural model
Overview: phases, activities, roles
STEP vs. common industry practices

Test Planning
Planning fundamentals
Planning and risk management
Master Test Plan–the IEEE 829 standard
Regression testing methods and issues
Test environments—issues and concerns
Schedules, estimates, and budgets
Acceptance, system, build/integration, and unit level test plans

Test Analysis
Identifying test objectives
Creating the inventory of test objectives
Determining the risk of each objective
Software failure modes
Estimating the testing effort based on risk
Testing system modifications, new versions, and third-party software
Analyzing requirements to develop test items

Test Design and Implementation
Creating an architecture for testing
Designing test cases and test procedures
Creating realistic test data
Developing reusable "testware"
Building the optimum test environment
Inventory trace matrix and coverage

Test Execution and Reporting
Executing tests
Reporting and managing defects
Performing root cause analysis
Evaluating the product and the development process
Evaluating the tests and the testing process
Choose appropriate "stopping" criteria

Final Perspective
Improving your testing process
Guidelines for fostering change
Course summary

ABOUT THE INSTRUCTOR

With more than thirty years of experience in the field of software development and testing, ***Lee Copeland*** *has worked as a programmer, development director, process improvement leader, and consultant. He has developed and taught many training courses focusing on software testing and development issues based on his experience and is the author of* A Practitioner's Guide to Software Test Design. *Lee is the Managing Technical Editor for* Better Software *magazine and is a regular columnist for StickyMinds.com.*

Additional instructors for this course include Dale Perry, Rick Craig, Claire Lohr, Robert Sabourin, Dawn Haynes, and Richard Bender.

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/sst

PUBLIC COURSE OFFERINGS

TW **New York/New Jersey Area**	September 8–10, 2008
TW **Washington, DC**	September 15–17, 2008
TW **San Francisco, CA**	October 20–22, 2008
TW **Tampa, FL**	November 17–19, 2008

TW *Indicates a Training Week course. See page 4 for details.*

SUGGESTED COURSES FOR PAIRING

Mastering Test Design
Requirements-Based Testing
Performance, Load, and Stress Testing
Lean-Agile Testing Practices
Test Process Improvement
Software Security Testing and Quality Assurance
Exploratory Testing in Practice

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

MASTERING TEST DESIGN UPDATED

Techniques for Developing Focused Test Cases

Project Management Institute R.E.P.

You will earn 15 contact hours or PDUs upon completion of this course.

- Learn functional, black-box test design techniques to find more bugs—faster
- Explore structural, white-box methods to add more depth to test cases
- Practice these test design techniques to reinforce your new skills
- Examine exploratory testing approaches to replace ad hoc testing
- Find out when to use each test design technique for the best results

Take-Home Bonus

Each public course participant receives a copy of *A Practitioner's Guide to Software Test Design*. Order additional copies by visiting www.sqe.com/books.asp

Valid for public courses only.

The Practical "How-To's" of Creating Test Cases

This course begins where many software testing courses end. Once the test plans are written, test teams are formed, and test tools are selected, it is time to create test cases. Since testing everything is impossible, the first step in test design is to choose a subset of all possible tests of program paths and data combinations to find important defects quickly. Mastering Test Design teaches you to select an optimal set of what to test and develops your practical skills to become a better test engineer.

Hands-On Practice of Testing Techniques

Experience the science and the art of both functional and structural testing methods in an informal workshop setting. First, you will see real-world examples of each test technique demonstrated. Then, you'll have the opportunity to practice these techniques, gain new testing skills, improve your testing effectiveness, and increase your professionalism. You will leave this class with a newfound confidence for developing test cases that find important bugs earlier.

For Software Developers and Testers

This course is appropriate for both novice and experienced software testers. Developers in the agile world who are now expected to write test cases will find this course extremely useful. Test and development managers can also benefit from this course. A background of basic development processes and test levels is helpful but not required.

This course is also available for Web-based eLearning. See page 21 for more information.

ABOUT THE INSTRUCTOR

Claire Lohr *has been a professional in the computer field for more than thirty years, with the last fifteen years focused on software process improvement for companies including GTE, Motorola, Westinghouse, SAIC, Boeing, Aetna, and others. Claire currently provides training and consulting services for a wide variety of both government and commercial clients. Her certifications are CSQE, CSDP, and CTFL. Claire is an SEI CMM Software Capability Evaluator and a Lloyd's Register ISO 9000 Lead Auditor.*

Additional instructors for this course include Lee Copeland, Dale Perry, Richard Bender, Robert Sabourin, Rick Craig, Bill Lewis, and Dawn Haynes.

PUBLIC COURSE OFFERINGS

New York/New Jersey Area	September 11–12, 2008
Washington, DC	September 18–19, 2008
San Francisco, CA	October 23–24, 2008
Tampa, FL	November 20–21, 2008

Indicates a Training Week course. See page 4 for details.

SUGGESTED COURSES FOR PAIRING

Systematic Software Testing
Test Management
Writing Testable Requirements
Just-in-Time Software Testing

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/mtd

2 Days Topical Outline:

Introduction

Place of test design within the testing process
Test case elements
Understanding test oracles
Test case selection - trade-offs

Functional—Black-Box Test Techniques

What is black-box testing?
Black-box testing at different testing levels
Equivalence class partitioning
- Discovering and documenting partitions
- Partitioning complex fields
- Equivalence classes for multiple requirements

Boundary value analysis
Understanding boundaries
Challenging boundary issues
- Grouping input data

Exercise
Decision tables
- Decision table construction
- Grouping variables
- Complex conditions
- Turning decision tables into test cases

State-transition diagrams and tables
- State notation
- Designing tests from state-transition diagrams

Pairwise test methods
- Orthogonal arrays
- Combinatorial analysis

Structural—White-Box Test Techniques

What is white-box testing?
Control flow concepts
- Applying control flow to code
- Understanding paths and cyclomatic complexity

Code coverage
- Unit statement and decision coverage
- Integration path analysis
- Coverage applied at other levels

Data flow analysis
- Data variable relationships

Exploratory and creative testing
Beyond formal and scripted testing
- Creative invalids
- Error guessing
- Group insights
- Exploratory testing process

SOFTWARE TESTING CERTIFICATION

Certified Tester—Foundation Level Training

- Basics of testing—goals and limits, risk analysis, prioritizing, completion criteria
- Testing in software development—unit, integration, system, acceptance, and regression testing
- Test management—strategies and planning, roles and responsibilities, defect tracking, and test deliverables

Are you looking for an internationally recognized certification in software testing? Delivered by top experts in the testing industry, Software Testing Certification is an accredited training course to prepare you for the ISTQB™ Certified Tester—Foundation Level exam. This program is the only internationally accepted certification for software testing, accredited by the ISTQB™ through its network of national boards. The ISTQB™, a non-proprietary organization, has granted more than 80,000 certifications in more than twenty countries around the globe.

In the Software Testing Certification training course, learn the basics needed to become a software test and quality assurance professional and understand how testing fits into software development. Find out what it takes to be a successful software test engineer and how testing can add significant value to software development.

Project Management Institute R.E.P.

You will earn 22.5 contact hours or PDUs upon completion of this course.

Who Should Attend

The Software Testing Certification training course is appropriate for individuals who recently entered the testing field and those currently seeking certification in software testing.

3 Days Topical Outline:

Fundamentals of Software Testing

Software context: Why does software fail?
Principles, scope, and focus of testing
Debugging vs. testing
Understanding risk
Risk analysis: prioritizing using risk analysis
Goals of testing
The basic testing process
Test psychology – viewpoints on testing

Testing Throughout Software Development

Testing and development
Early testing
Models and testing
The "V" model
Verification and validation
Testing levels/stages within software development
Understanding regression testing
Understanding test types

Static Techniques

What is static testing?
Reviews, inspections, walkthroughs, etc.
General review process
Common types of reviews
Roles and responsibilities in reviews
Success factors for reviews
Limits of reviews
Understanding static analysis tools

Test Design Techniques

Overview of test design and the design approach
Documentation decision
Types of test design techniques
Human/experienced-based methods
Black-box (functional) techniques
White-box (structural) techniques
Experienced-based techniques
Selecting the appropriate test technique

Test Management

Team organization
Roles and responsibilities
Understanding the test manager
Understanding the tester
Test planning and strategy
Configuration management and testing
Defect/incident classification and management

Tool Support for Testing

Selection process
Introduction
Benefits
Risks and concerns
Classifications

At the conclusion of the course, you will have the opportunity to take the ISTQB™ Certified Tester—Foundation Level exam. The exam is held at 3:30 p.m. on the third day of the course. The ISTQB™ Certified Tester—Foundation Level certification exam is independently administered by the American Software Testing Qualifications Board.

A $250 fee for the exam is added to your registration fee. For more information on ISTQB™ certification or to download the syllabus, please visit ***www.astqb.com.***

ABOUT THE INSTRUCTOR

Dale Perry *has more than thirty years of experience in information technology. He has been a programmer/analyst, database administrator, project manager, development manager, tester, and test manager. Dale's project experience includes large systems development and conversions, distributed systems, on-line applications, both client/server and Web based. He has also been a professional instructor more than fifteen years and has presented at numerous industry conferences on development and testing. With Software Quality Engineering for eleven years, Dale has specialized in training and consulting on testing, inspections and reviews, and other testing and quality related topics.*

Additional instructors for this course include Claire Lohr, Dawn Haynes, Rick Craig, Robert Sabourin, Eric Patel, Jamie Mitchell, Gary Mogyorodi, and Ed Weller.

SUGGESTED COURSES FOR PAIRING

Performance, Load, and Stress Testing
Lean-Agile Testing Practices
Requirements-Based Testing
Test Process Improvement
Software Security Testing and Quality Assurance
Exploratory Testing in Practice

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

PUBLIC COURSE OFFERINGS

	Boston, MA	August 26–28, 2008
TW	**New York/ NJ**	September 8-10, 2008
	Minneapolis, MN	September 9–11, 2008
	Salt Lake City, UT	September 9–11, 2008
TW	**Washington, DC**	September 15–17, 2008
	Atlanta, GA	September 23–25, 2008
	Philadelphia, PA	September 23–25, 2008
	Anaheim, CA	September 28–30, 2008
	Indianapolis, IN	September 30–October 2, 2008
	Jacksonville, FL	October 7–9, 2008
	Toronto, ON	October 7–9, 2008
	Kansas City, MO	October 14–16, 2008
	Rochester, NY	October 14–16, 2008
TW	**San Francisco, CA**	October 20–22, 2008
	Pittsburgh, PA	October 21–23, 2008
	Sacramento, CA	October 21–23, 2008
	Charlotte, NC	October 28–30, 2008
	Omaha, NE	October 28–30, 2008
	Cincinnati, OH	October 28–30, 2008
	Ft. Lauderdale, FL	November 4–6, 2008
	Bethesda, MD	November 4–6, 2008
TW	**Tampa, FL**	November 17–19, 2008
	Sunnyvale, CA	November 18–20, 2008
	Phoenix, AZ	December 2–4, 2008

TW *Indicates a Training Week course. See page 4 for details.*

ALP *Bethesda, MD courses are held in cooperation with ALP International.*

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/stf

LEAN-AGILE TESTING PRACTICES NEW

Rapid Delivery of High Quality Software

- Apply lean principles to quality and testing
- Deliver value to customers quickly with agile testing practices
- Discover opportunities for lean-agile improvements
- Learn about lean-agile testing practices across the entire release
- Evaluate FitNesse as an automated acceptance testing tool solution
- Respond to and recommend agile testing transition actions

Quickly Deliver the Highest Value Features to the Customer

Lean-agile methods promote the rapid delivery of value to customers by deferring detailed definition and design of system features until the "last responsible moment." This practice challenges the whole team, including testing, to stay continuously synchronized within very short release iteration cycles. To keep up with the fast pace, the team must be creative, smart, and efficient with their verification and validation testing activities.

Lean Principles Add Value to Your Organization

In this interactive workshop, you will examine agile testing practices including exploratory testing and automated acceptance testing approaches. Experience the "four-hour release" cycle and practice the activities important to testers in a lean-agile development environment.

Adapt to Agile Development Practices that Affect Your Team

Teams new to agile practices have discovered that moving from traditional "test last" to the lean-agile "test first" is a big challenge to the entire development organization—and especially to testers and the test group. Discuss the common obstacles facing teams and explore solutions that can work for your test team. Develop an action plan to become valued members of a lean-agile development team.

Who Should Attend

The audience includes test professionals, test managers, project leaders, quality analysts, and developers. No specific prerequisites are assumed. Experience in any form of software development (testing, programming, and managing) and some familiarity with agile practices are recommended.

ABOUT THE INSTRUCTOR

Bob Hartman *has more than thirty years of experience developing software, including seven years running his own consulting company and more than fourteen years of experience at the VP of Development level or higher. Bob has served in every role in the software industry including developer, tester, documentation writer, trainer, manager, and executive. A Certified Scrum Master, Bob does training and coaching in agile development. Bob teaches courses including Lean Agile Testing, Implementing Scrum, and Lean Software Development. He also has a unique talent for breaking software within the first ten minutes of using it.*

PUBLIC COURSE OFFERINGS

TW **New York/New Jersey Area**	September 11–12, 2008
TW **Washington, DC**	September 18–19, 2008
TW **San Francisco, CA**	October 23–24, 2008
TW **Tampa, FL**	November 20–21, 2008

TW *Indicates a Training Week course. See page 4 for details.*

SUGGESTED COURSES FOR PAIRING

Writing Testable Requirements
Systematic Software Testing
Software Testing Certification
Just-in-Time Software Testing
Test Management

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

2 Days Topical Outline:

The Power of Lean-Agile Development

Seven principles of lean-agile development
Test-driven design and unit testing
Continuous test-driven integration
Lean-agile quality planning
Exercise

How Agile Changes Requirements and Testing

Requirements definition—use cases, user stories
Agile testing vs. traditional testing
Test artifacts for lean-agile development
Manual vs. automated testing
Agile team case study
Mike Cohn's testing pyramid
Exercise

Agile Testing Practices for Speed of Need

Exploratory testing
Quick explore
Blink
Session-based testing
Release explore
Exercise

Automated Acceptance Testing

Improve acceptance testing
Open source FitNesse
Exercise
Agile Testing Activities throughout a Release

Agile Testing Transitions

People challenges—discussions and suggestions
Exercise
Technical challenges—discussions and suggestions
Exercise
Organization challenges—discussions and suggestions
Exercise

Look Back, Look Ahead

Where do we start? Where are we now?
Where do we go for help?
What are your next steps?

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/atp

3 Days Topical Outline:

Be Prepared–What You Need
Bug tracking
Test oracles
Operational profiles
Test environment

Testing Ideas–What To Test
Usage scenarios and data
Requirements and design documents
Failure modes
Capabilities and domains
Quality factors
Creative techniques
Exercise—Creative test idea generation

What Not to Test
Consequences and benefits of skipping
Consequence of implementing
Refactoring tests
Credibility of tests

Testing Triage
Roles and responsibilities
Triage concerns
Triage through project lifecycle
Adapting to project context for triage
Exercise—Testing triage practice session

Regression
Did we really fix the bug?
Did we accidentally break something?

Testing in the Development Lifecycle
Requirements workflow
Bug workflow
Configuration management
Iterative, agile, and RUP approaches

Measurement and Reporting
When tests should be run again
Elaboration states
Metrics and coverage
Exercise—Interpreting real-world status reports

Session-Based Exploratory Testing
Balancing scripted tests vs. exploration
Exploratory testing sessions
Test charters
Testing notes
Building the exploration map
Accountability
Exercise—Hands-on exploratory testing session

Just Enough Test Automation

JUST-IN-TIME SOFTWARE TESTING NEW

Powerful Tools for Fast-Changing Projects and Priorities

- Test projects that have few or no written requirements
- Conduct testing "triage" to find important bugs more quickly
- Learn to plan and schedule testing in a dynamic, unpredictable world
- Practice session-based exploratory testing to find show-stopper bugs and change the way you test
- Gain the confidence you need to succeed
- Learn to blend exploratory, scripted, and automated testing

Dealing with Software Project Turbulence

Turbulent development projects experience almost daily requirements changes, user interface modifications, and the continual integration of new functions, features, and technologies. Keep your testing efforts on track while reacting to changing priorities, technologies, and user needs. This highly interactive workshop offers a unique set of tools to help you cope with—and perhaps even flourish in—what may seem to be a totally chaotic environment. Practice dynamic test planning and scheduling, test idea development, bug tracking, reporting, test triage, exploratory testing, and much more.

Getting Ready for Almost Anything They Can Throw at You

Be ready for just about anything that can happen in a software testing project such as a complex, customer-facing Web or e-commerce application. Learn to identify, organize, and prioritize your testing "ideas." Create workflows to schedule testing tasks dynamically, conduct bug triage sessions, and adapt the testing focus as priorities change. Decide on purpose what not to test—not just because the clock ran out!

Real Techniques Proven in Real Projects

Just-In-Time Testing (JIT) approaches are successfully applied to many types of software projects—commercial off-the-shelf applications, agile and iterative development environments, mission-critical business systems, and just about any Web application. Real examples demonstrate how JIT testing either replaces or complements more traditional approaches. Examples are drawn from insurance, banking, telecommunications, medical, and other industries. The course is packed with interactive exercises in which students work together in small groups to apply JIT testing concepts.

Who Should Attend

This course is appropriate for anyone who works in fast-paced development environments, including test engineers, test managers, developers, QA engineers, and all software managers.

ABOUT THE INSTRUCTOR

Robert Sabourin *has more than twenty-five years of management experience, leading teams of software development professionals. A well-respected member of the software engineering community, Robert has managed, trained, mentored, and coached hundreds of top professionals in the field. He frequently speaks at conferences and writes on software engineering, SQA, testing, management, and internationalization. The author of* I am a Bug!*, the popular software testing children's book, Robert is an adjunct professor of Software Engineering at McGill University.*

Scott Barber is an additional instructor for this course.

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/jit

PUBLIC COURSE OFFERINGS

TW **Washington, DC**	September 15–17, 2008
TW **San Francisco, CA**	October 20–22, 2008
TW **Tampa, FL**	November 17–19, 2008

TW *Indicates a Training Week course. See page 4 for details.*

SUGGESTED COURSES FOR PAIRING

Lean-Agile Testing Practices
Mastering Test Design
Requirements-Based Testing
Performance, Load, and Stress Testing
Test Process Improvement
Software Security Testing and Quality Assurance
Exploratory Testing in Practice

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

SOFTWARE SECURITY TESTING AND QUALITY ASSURANCE NEW

An Integrated, Risk-Based Approach to More Secure Software

- Learn practical security testing and QA approaches based on experiences of top software security experts
- Use a structured risk analysis method to expose dangerous software vulnerabilities
- Learn how to add security testing to all of your testing strategies and plans
- Discover how hackers exploit software so you can "think like an attacker" while developing your tests
- See real-life examples of severe security defects at both the design and implementation levels
- Integrate security testing and QA practices into your entire software development lifecycle

cigital

Ensure Your Software is Secure

Developed by top experts in software security, this course is an eye-opening experience for all test and QA professionals and for development professionals doing structured unit and integration testing. It will change the way you think about test development.

Proactive security testing requires that you first get an understanding of the security problem and adopt a risk management framework for addressing security issues. Then, you need to gain the skills and implement the processes necessary to develop and execute security test strategies.

Think Like an Attacker and Test Like One

Learn to think like an attacker so that you can add test cases to cover non-functional—often implied or missing—security requirements. Find out about the "Seven Pernicious Kingdoms of Software Security" and how to use security risk information to improve test and QA strategies and planning. Practice examining software requirements, designs, and code to expose security vulnerabilities as early as possible during development. Add appropriate abuse cases to your test designs and explore your software with a new awareness of security issues.

Put Risk-Based Security Testing into Practice

Look inside the code with white-box testing techniques to achieve greater benefits with less effort. Tie in the business and design objectives, architectural and operational realities, and common attack patterns to enhance your current testing methods. With new knowledge and skills, you can build the confidence that attackers cannot turn security risks into security failures.

Who Should Attend

This course is appropriate for testing, QA, and software development practitioners who are responsible for developing and executing test strategies and plans for functional and non-functional security requirements. This course requires an ability to understand security risk patterns used by attackers. Participants should be comfortable reviewing code as part of their testing activities.

ABOUT THE INSTRUCTOR

Paco Hope *is a managing consultant at Cigital and has more than twelve years of experience in software security and operating system security. His areas of expertise include software security policy, code analysis, host security, and PKI. Paco has worked extensively with embedded systems in the gaming and mobile communications industries, and also has served as a subject matter expert on issues of network security standards in the financial industry. Paco is co-author of* Mastering FreeBSD and OpenBSD Security, *published by O'Reilly and Associates, and has published articles on abuse cases in software design, PKI, and UNIX host security features. Prior to joining Cigital, he served as director of product development for Charlottesville, VA-based Tovaris, Inc. and head systems administrator in the Department of Computer Science at the University of Virginia.*

Richard Mills is an additional instructor for this course.

PUBLIC COURSE OFFERINGS

TW **Washington, DC**	September 18–19, 2008
TW **San Francisco, CA**	October 23–24, 2008

TW *Indicates a Training Week course.* *See page 4 for details.*

SUGGESTED COURSES FOR PAIRING

Writing Testable Requirements
Systematic Software Testing
Software Testing Certification
Just-in-Time Software Testing
Test Management

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/stq

2 Days Topical Outline:

Software Security and Testing
The software security problem
What is security testing?
Exploiting software
Attack patterns
Exercise: Risk analysis

Software Security Touchpoints
Risk management framework
Software security touchpoints

Seven Pernicious Kingdoms
Classic "gotchas"
Seven kingdoms of security errors
Exercise: Code review—seven kingdoms

Static Analysis and Code Review
Static analysis and code review
Tool demo

Software Security Requirements Analysis
Nonfunctional requirements
Negative requirements
Use and abuse cases

Security Testing Overview
Process overview
Learning from history
Goals, principles, and methods

Risk-Based Security Testing Process
Risk assessment results
Abuse cases
Test strategy and planning

Examples from the Seven Pernicious Kingdoms
SQL injection
Cross-site scripting
Return values
Violation of trust levels
Many others

Putting Risk-Based Security Testing into Practice
Tying it all together
Software security framework
Software security roadmap

Take-Home Bonus

Each public course participant receives a copy of Gary McGraw's *Software Security: Building Security In.*

Valid for public courses only.

WRITING TESTABLE REQUIREMENTS

Produce Better Requirements to Reduce Effort and Cost

- Recognize "good" and "bad" requirements
- Correct ambiguities in specifications to reduce or eliminate re-work and to make testing more effective
- Deliver requirements that are concise, accurate, modular, and highly testable

3 Days Topical Outline:

Why Good Requirements Are Critical
Impact on costs of development
Impact on schedules

Characteristics of Good Requirements

Key Characteristics for Testability

Industry Guidelines for Requirements
IEEE STD-830-1998
UML

Identifying Ambiguities in Specifications

Defining Clear Objectives and Problem Statements

Ensuring Commonality
Common expectations
Common language
Common "world view"

Writing Style Guidelines for Describing Processes/Use Cases
Style suggestions for readability
Describing decision logic and transforms
Impact of physical design
Structured English vs. pseudo code

Process Packaging Guidelines

Writing Style Guidelines for Describing Data

Guidelines for Naming Processes and Data

Automated Requirements Management

Focus Up-Front on Problem Avoidance

Inferior requirements significantly increase the cost of system development and the time required to deliver an application. Writing Testable Requirements focuses on problem avoidance—how to write requirements accurately the first time or to improve them before coding starts. This training course offers guidelines for describing software specifications of processes and data, ensuring that requirements have the clarity and detail needed to produce test cases.

This information is critical to designers, developers, and technical writers. The techniques can be applied to requirements written to various company or industry standards. The course also addresses compliance with common industry guidelines and the effects of automated repositories on requirements writing styles. You are encouraged to bring samples from your own projects to work on and evaluate during class.

For Analysts, Testers, Developers, and Managers

This course is intended to help those who write and review detailed functional specifications and those who must develop and test systems based on those requirements. The intended audience includes testing staff, requirements analysts, developers, and project managers. No specific prerequisites are assumed, but you are expected to be software knowledgeable with at least basic test or software development experience. Requirements-Based Testing (page 15) is a valuable complement to this course. Take both courses during a Training Week to save time and money!

ABOUT THE INSTRUCTOR

Richard Bender *has been involved in test and evaluation since 1969. He has authored and co-authored books and courses on quality assurance and test, software development lifecycles, analysis and design, software maintenance, and project management. Richard has worked with an international clientele in a wide range of industries from financial to academic.*

Gary Mogyorodi and Bill Lewis are additional instructors for this course.

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/wtr

PUBLIC COURSE OFFERINGS

TW **Washington, DC**	September 15–17, 2008
TW **San Francisco, CA**	October 20–22, 2008
TW **Tampa, FL**	November 17–19, 2008

TW *Indicates a Training Week course. See page 4 for details.*

SUGGESTED COURSES FOR PAIRING

Requirements-Based Testing
Mastering Test Design
Performance, Load, and Stress Testing
Lean-Agile Testing Practices
Test Process Improvement
Software Security Testing and Quality Assurance
Exploratory Testing in Practice

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

REQUIREMENTS-BASED TESTING

A Disciplined Approach for Testing—and Improving—Software Requirements

- Identify important ambiguities in requirements specifications before coding starts
- Translate requirements specifications into cause-effect graphs to verify accuracy and completeness
- Design a set of test cases to validate that all requirements are implemented
- Quantify and accurately measure the progress of your testing efforts

Get a Realistic Picture

Testing, by definition, compares an expected result to the observed result. In software, the expected results should be defined in the specifications. Unfortunately, most specifications are not sufficiently detailed to define the expected results. This process-oriented course presents a set of practical, yet rigorous, techniques for testing requirements to ensure that your project's requirements are complete, consistent, accurate, and unambiguous.

What to Expect

Once the specifications have been clarified, the second challenge is to define the necessary and sufficient set of tests to verify that the design and code fully meet the specifications. The Requirements-Based Testing (RBT) course teaches you how to design a consistent and repeatable set of test cases.

Using RBT, test completion criteria are quantified and test status is measurable. RBT provides a process for first testing the integrity of the specifications. It then provides the algorithms for designing an optimized set of tests sufficient to verify the system from a black-box perspective.

Who Should Attend

This student-paced course is designed for test engineers, quality assurance engineers, and software managers. You are encouraged to bring samples from your own projects. The focus of the course is on process tools. However, there's a brief introduction to using BenderRBT™, which automates much of the requirements-based testing process. Writing Testable Requirements or Finding Ambiguities in Requirements is a prerequisite for this class.

ABOUT THE INSTRUCTOR

Richard Bender *has been involved in test and evaluation since 1969. He has authored and co-authored books and courses on quality assurance and test, software development lifecycles, analysis and design, software maintenance, and project management. Richard has worked with an international clientele in a wide range of industries from financial to academic.*

Gary Mogyorodi and Bill Lewis are additional instructors for this course.

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/rbt

PUBLIC COURSE OFFERINGS

TW **Washington, DC**	September 18–19, 2008
TW **San Francisco, CA**	October 23–24, 2008
TW **Tampa, FL**	November 20–21, 2008

TW *Indicates a Training Week course.* *See page 4 for details.*

SUGGESTED COURSES FOR PAIRING

Writing Testable Requirements
Systematic Software Testing
Software Testing Certification
Just-in-Time Software Testing
Test Management

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

2 Days Topical Outline:

Introduction
Definition of testable requirements
Definition of testing—the seven basic steps
The business case for quality
Overview of the RBT process

Initial RBT Steps
Validating requirements against objectives
Validating the scope of requirements via scenarios
Cause-effect graphing
Basic logical operators
Five graphing constructs of all functional requirements

Environmental Data Constraints
Boundary condition data constraints
Processing sequence imposed constraints
Identifying logical inconsistencies in the processing rules
Test-case design

Defining the Test Completion Criteria
Concept of fault detection
Basic strategies for test case design
Identifying functional variations
Packaging functional variations into test cases
Equivalence class testing and orthogonal pairs

Additional Points of Integration
Moving into the development process
Moving integration testing up before coding starts
Moving use acceptance testing up before coding starts

Introduction to Code-Based Testing
White-box test completion criteria
Data-flow-based testing
Integrating black-box and white-box testing

Tuning the RBT Process by Project Type
Rapid prototyping
Rapid application development
Agile methodologies
New development
Third-party packages
Maintenance
Technology conversions
Rewrites and re-engineering projects

Management Considerations
Planning and estimating guidelines
Change control
Impact on staffing
Test team organization
Tracking the testing effort
Contract management

Course Wrap-Up

2 Days
Topical Outline:

Fundamentals
Imperative to performance test
Performance testing track record

The Performance Testing Process
Understanding how performance testing fits the development process
Approaches to the performance testing process
Costs of performance testing

Identify Performance Goals and Business Goals
Gather background information on the situation
Develop an understanding of the situation
Validate the test project need and feasibility

Gain an Understanding of the Infrastructure and Architecture Required for the Test
What must be part of the test?
What can be omitted?

Key Areas of Infrastructure/ Architecture
Target platform and systems
Network configuration
Performance tools
Understanding scalability and extrapolation

What Types of Tests and Measurements Do We Need?
Types of performance tests to be run

Understanding and Defining Workload
Transactions to be simulated
Analyze factors affecting the load definition

Understanding Key Measurements
Response times, resource usage, etc.

Understanding Test Preparation
Setting up the test infrastructure/architecture
Acquire the test scripts and data
Setting up the tools

Execute the Tests
Validate the tests and the tools
Prepare for the test execution
Execute the tests and collect the data
Analyze the data and evaluate the test results
Present conclusions and recommendations
Assist the technical team after tuning and debugging

PERFORMANCE, LOAD, AND STRESS TESTING

Issues and Solutions for Software Performance Testing

- Understand the performance testing process
  - –Test planning
  - –Test preparation
  - –Test execution
  - –Reporting test results
- Relate performance testing to the development process
- Understand performance goals and objectives
- Learn how to deal with environment and architecture issues
- Understand and select the various types of performance tests
- Define operational profiles and load definitions
- Define and select appropriate measurements

In the Real World

This practical, hands-on course delivers testing skills that participants can immediately apply back on the job. Using a real-world case study, you will encounter issues, decisions, and testing experiences comparable to those in your own work environment. Working through a series of exercises in small teams, or as a group, you develop a workable strategy for performance testing and application/system.

Who Should Attend

System testers, system designers, system tuners, software engineers, quality assurance professionals, and project leaders who are involved in systems testing can benefit from this course. A working knowledge of system testing and quality assurance fundamentals is assumed, but no specific technical background (e.g., UNIX, TCP/IP) is required.

ABOUT THE INSTRUCTOR

Dale Perry *has more than thirty years of experience in information technology. He has been a programmer/analyst, database administrator, project manager, development manager, tester, and test manager. Dale's project experience includes large systems development and conversions, distributed systems, on-line applications, both client/server and Web based. He has also been a professional instructor for more than fifteen years and has presented at numerous industry conferences on development and testing. With Software Quality Engineering for eleven years, Dale has specialized in training and consulting on testing, inspections and reviews, and other testing and quality related topics.*

Scott Barber is an additional instructor for this course.

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/plt

PUBLIC COURSE OFFERINGS

TW **Washington, DC**	September 18–19, 2008
TW **San Francisco, CA**	October 23–24, 2008
TW **Tampa, FL**	November 20–21, 2008

TW *Indicates a Training Week course. See page 4 for details.*

SUGGESTED COURSES FOR PAIRING

Systematic Software Testing
Software Testing Certification
Writing Testable Requirements
Just-in-Time Software Testing
Test Management

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

EXPLORATORY TESTING IN PRACTICE NEW

Plan, Design, and Execute Tests Simultaneously to Find More Bugs—Faster

- Make real time decisions for better application testing
- Design your tests as you perform them
- Organize exploratory testing for you and your team
- Practice a session-based framework for exploratory testing
- Implement test charters to guide your testing
- Examine examples and real world case studies

Many traditional test teams are augmenting their documented test plans and test cases with a structured, exploratory approach. Other teams adopting agile methods are replacing ad-hoc testing with exploratory techniques, allowing all development team members to effectively participate in product testing. Whether your organization is moving toward agile software practices or using a more traditional approach, exploratory testing can help you find important defects sooner.

Exploratory testing is all about simultaneously learning about the software you are testing while you are designing and executing the tests. It is used by developers for unit testing, independent testing teams for integration or system testing, and by customers implementing acceptance testing of developed or commercial off-the-shelf software packages.

In this highly interactive class, students learn about and practice session-based exploratory testing, a framework to organize testing into a series of time boxed missions or "charters." In fulfilling a test charter, you use your skills and experience to adapt your testing actions as you learn what the application does. Through this process, one discovery leads to another and another as you explore the software under test. Exploratory testers add permanent value to projects by constructing practical notes, which provide short valuable logs that record what was discovered during each testing session.

Through a series of small group, hands-on exercises, students practice exploratory testing and improve their skills as they test. In addition, you will learn how and when to use exploratory testing practices in different project and organizational contexts. Review the tools that are available to organize and support exploratory testing, and capture data from exploratory testing sessions. Return to your team with new skills and processes to make your testing more effective—and more fun.

Who Should Attend

This course is appropriate for anyone who works in fast-paced testing environments, including test engineers, test managers, agile developers, QA engineers, and all software managers. Customers charged with acceptance testing and traditional unit testers will also benefit from the course.

Each participant in this course is required to bring a Windows laptop computer.

ABOUT THE INSTRUCTOR

Robert Sabourin *has more than twenty-five years of management experience, leading teams of software development professionals. A well-respected member of the software engineering community, Robert has managed, trained, mentored, and coached hundreds of top professionals in the field. He frequently speaks at conferences and writes on software engineering, SQA, testing, management, and internationalization. The author of* I am a Bug!*, the popular software testing children's book, Robert is an adjunct professor of Software Engineering at McGill University.*

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/et

PUBLIC COURSE OFFERINGS

Washington, DC	September 18–19, 2008
San Francisco, CA	October 23–24, 2008
Tampa, FL	November 20–21, 2008

Indicates a Training Week course. See page 4 for details.

SUGGESTED COURSES FOR PAIRING

Writing Testable Requirements
Systematic Software Testing
Software Testing Certification
Just-in-Time Software Testing
Test Management

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

2 Day Topical Outline:

Overview
Definitions
History
Styles
Strengths and weaknesses
Case studies
Lifecycle models
Context drivers

Getting Organized
Sessions
Charters
Focus and opportunity
Measures

Testing Skills
Observation
Reasoning
Test design
Failure analysis
Pivoting
Note taking

Tools
Capturing test data
Note taking and mind mapping
Test design
Combination testing
Test frameworks

References
Articles
Books
Courses
Web resources

Exercise Outline
Class exercises are used to illustrate concepts covered in the class. Students will work in teams of two, using open source tools for the exercises. We encourage you to bring applications from your work environment to use for some of the exercises.

Capabilities and instabilities
What can the application do?
Areas of weakness?
What can break?

Variables and emergent behaviors
Discover variables
Explore emergent behaviors
Influencers and outcomes
Test oracles

Usage scenarios
Who uses the software?
What do they do?
Can it be done?

Back to back testing
Side-by-side comparison
Discovering differences
Confirming capabilities

Test design and visual modeling on the fly
Equivalence partitioning and boundaries
Business logic
Decision tables

Exploring quality factors
Performance testing
Load testing
Stress testing

3 Days Topical Outline:

Testing and Quality
Goals vs. priorities
Testing philosophy
Preventive testing and the STEP™ methodology

The Test Manager
Roles of a test manager
Leadership guidelines
Ways to obtain buy-In

Test Teams
Team organization
Retaining staff
Staff development

Configuration Management
Library management
Change control board
Defect analysis

Master Test Plan
Risks and contingencies
Testing strategy
Scheduling/estimating

Testware Design
Influence of risk
Design approaches
Testing without requirements

Execution Management
Assessing test effectiveness
Predicting release dates
Reporting

Tools
What to automate
Tool issues
Manager's role in tool implementation

Metrics Primer
Basic definitions
Measurement paradigms
Implementing a metrics program

Process Benchmarks and Baselines
What is process assessment?
ISO, CMM®, TQM, TPI®
Benchmarking

TEST MANAGEMENT

What Every Test Manager Needs to Know

- Identify and handle critical strategy issues in specific test situations
- Monitor the status of testing activities and software work products
- Work productively with users, developers, and support staff

The Importance of Strategy and Feedback

If you develop and implement an effective test strategy, you can successfully manage software test efforts. Successful test management requires the same approach as successful project management: 1) develop a sound strategy, 2) keep in close touch with the situation, 3) identify and aggressively manage critical issues, and 4) modify the strategy as needed, based on situational feedback. The key to test management is to know the components of an effective test strategy, including feedback mechanisms, and to recognize critical issues as they surface.

A Management Framework

This course provides the essential framework for successful test management. It focuses on two critical areas: 1) creation and management of a successful testing organization and team, and 2) development of an effective test strategy. This strategy is built around the development of two key documents: a comprehensive test plan and corresponding test report.

For Anyone Responsible for Software Test Efforts

This course provides the information necessary for a test manager, supervisor, or senior tester to lead a testing effort. It also provides an outline of good testing methods from the viewpoint of a test manager. This course complements the Systematic Software Testing course, which provides a similar outline of good testing methods from the viewpoint of the test engineer. Participants should have at least six months of leadership experience and two years of test experience.

Take-Home Bonus

Each public course participant receives a copy of *Systematic Software Testing*. Order additional copies by visiting www.sqe.com/books.asp

Valid for public courses only.

ABOUT THE INSTRUCTOR

A frequent speaker at testing conferences, **Rick Craig** *is recognized worldwide as a test and evaluation instructor with SQE Training. He has implemented and managed testing efforts on large-scale, traditional, and embedded systems, and co-authored a study that benchmarked industry-wide processes. Rick is co-author of the reference book* Systematic Software Testing.

Lee Copeland is an additional instructor for this course.

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/tm

PUBLIC COURSE OFFERINGS

TW **Washington, DC**	September 15–17, 2008
TW **San Francisco, CA**	October 20–22, 2008
TW **Tampa, FL**	November 17–19, 2008

TW *Indicates a Training Week course. See page 4 for details.*

SUGGESTED COURSES FOR PAIRING

Lean-Agile Testing Practices
Mastering Test Design
Requirements-Based Testing
Performance, Load, and Stress Testing
Test Process Improvement
Software Security Testing and Quality Assurance
Exploratory Testing in Practice

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

TEST PROCESS IMPROVEMENT

Practical Guidelines for Small and Large Test Organizations

- Improve testing processes in any size test group
- Make visible step-by-step improvements for a fast payback
- Implement improvements in conjunction with or independent of the Capability Maturity Model (CMM® and CMMI®)

Get a Realistic Picture

Testing is often seen as a costly and uncontrolled process. Management often says that testing takes too much time, costs more than planned, and offers little insight into the quality of the system under test. If production systems are of poor quality, improving your testing process may help solve the problem.

Whether your test group consists of two people or two hundred, you'll take away from this workshop a proven, systematic approach to assess your current test processes and chart a course for measurable test improvement.

Approach

The way to improve the performance of your test group depends on the size and type of your test group, your software development lifecycle, and the maturity of your test team. In this interactive, workshop-style class, you'll learn how to customize the TPI® model to fit your organization and its needs.

You'll learn how to create awareness, establish goals and scope for change, perform the assessment, select prioritized improvement actions, and implement change. Get the practical guidelines on how to take each step, including references to where the test improvement model should be applied. You'll also learn about critical subjects needed for a successful improvement program, such as the use of metrics, the requirements for the change team, and dealing with resistance.

This interactive course offers a pragmatic, day-to-day improvement approach, including an outline for small test improvements, hints and tips for a quick start test improvement, and ways to achieve and maintain management buy-in. Potential pitfalls and expected benefits will be covered as well.

Who Should Attend

This course provides valuable information for software managers, test managers, test team leads, test consultants, QA managers, and IT process improvement specialists.

ABOUT THE INSTRUCTOR

A frequent speaker at testing conferences, **Rick Craig** *is recognized worldwide as a test and evaluation instructor with SQE Training. He has implemented and managed testing efforts on large-scale, traditional, and embedded systems, and co-authored a study that benchmarked industry-wide processes. Rick is co-author of the reference book* Systematic Software Testing.

Lee Copeland and Martin Pol are additional instructors for this course.

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/tpi

PUBLIC COURSE OFFERINGS

TW	**Tampa, FL**	November 20, 2008

TW *Indicates a Training Week course.* *See page 4 for details.*

SUGGESTED COURSES FOR PAIRING

Systematic Software Testing
Software Testing Certification
Writing Testable Requirements
Just-in-Time Software Testing
Test Management

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

1 Day Topical Outline:

Introduction

Typical complaints about testing
Improving the testing process
Position and scope of test process improvement
The process of change

Improvement Process

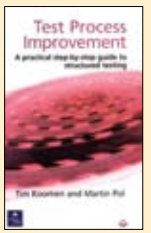
Awareness
Goal, scope, and approach
Assessment
Define improvement actions
Plan
Implementation
Evaluation

The Test Process Improvement (TPI®) Model

Relationship to CMM®
Assessment
Twenty key areas
Levels
Checkpoints
Improvement suggestions
The Test Maturity Matrix
Sequence of improvements

Improvement Actions

Define
Plan
Implement
Evaluate



Take-Home Bonus

Each public course participant receives a copy of the book, *Test Process Improvement: A Practical Step-by-Step Guide to Structured Testing*, by Tim Koomen and Martin Pol.

Valid for public courses only.

VISUAL STUDIO® 2008 TEAM FOUNDATION SERVER

Understanding and Administering Microsoft® Team Foundation Server

- Create and modify work items
- Understand how to administer and manage Team Foundation Version Control
- Customize work item definitions to support your custom workflow
- Create, execute, and modify custom build scripts for Team Build

This course covers the major components of Visual Studio® 2008 Team Foundation Server. Learn how to build your organization's software process into Team Foundation Server by modifying the out of the box process templates for MSF for Agile Software Development and MSF for CMMI® Process Improvement. This course covers version control, reporting, automated build, work item tracking, and project management.

Who Should Attend

This course is designed for individuals responsible for source control, project management, build administration, and reporting.

Bonus: Computer lab is supplied for this course.

In cooperation with: Notion

Microsoft CERTIFIED Partner

INSTRUCTORS: Chris Menegay, Dave McKinstry, Donovan Brown, and Trent Nix

PUBLIC COURSE OFFERINGS

Chicago, IL	September 22–23, 2008
Washington, DC	October 13–14, 2008

Indicates a Training Week course. See page 4 for details.

SUGGESTED COURSES FOR PAIRING

Testing Applications with Visual Studio® Team System 2008

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/vfs

2 Days Topical Outline:

Team System Overview
Development challenges and features
Role-based approach
Packaging
Team Foundation security settings
Groups, permissions, and supporting services
LAB

Team Projects and Process
How process is integrated with Visual Studio® 2008 Team System
MSF for Agile Software Development and CMMI® Process Improvement

Team Project Security
TFS security review, team project security, supporting services, and version control security
LAB

Project Management and Work Items
Work item tracking system
Creating custom work item queries
Work item integration with Microsoft Project and Microsoft Excel
LAB

Version Control
Architecture overview
Integrated and atomic check-in
Parallel and remote development
Check-in notes, notification, and policy
LAB

Advanced Version Control
Shelving, branching, and merging
Multiple solutions and common admin tasks
LAB

Build Process
Team build architecture
Creating build scripts
Executing builds and command-line tasks
Running Web tests
Scheduling builds
Continuous integration

Modifying Work Item Definitions
Changing work item definitions on live projects
Adding workflow and controlling field access
LAB

Modifying Process Templates
Modifying document templates, work items, and MS Project synchronization
Using custom processes with team projects
LAB

Planning Deployment
Logical architecture
Server deployment architecture
Common administrative tasks
Building reports
LAB

TESTING APPLICATIONS WITH VISUAL STUDIO® TEAM SYSTEM 2008

Managing Lifecycle Testing

- Understand how to create and manage a variety of tests using the testing tools in Visual Studio® 2008 Team System
- Learn how to enter and edit defects using work item tracking
- Find out how to analyze build reports and test results
- Become familiar with Team Foundation Version Control, where all tests are stored and historical changes are tracked

This course provides hands-on experience with all the Team System testing functions including: defect reporting, defect tracking, and manual test creation, as well as execution of manual, Web, load, and unit tests. It covers how to utilize reporting features and create quality reports used to analyze the status of projects.

The testing portion of this course is taught using a shared Team Foundation Server. The testing aspects utilize Visual Studio® Team System 2008 Test Edition.

Who Should Attend

This class is intended for those individuals responsible for the QA and testing roles in software development.

Bonus: Computer lab is supplied for this course.

In cooperation with: Notion

Microsoft CERTIFIED Partner

INSTRUCTORS: Chris Menegay, Dave McKinstry, Donovan Brown, and Trent Nix

PUBLIC COURSE OFFERINGS

Chicago, IL	September 24–25, 2008
Washington, DC	October 15–16, 2008

Indicates a Training Week course. See page 4 for details.

SUGGESTED COURSES FOR PAIRING

Visual Studio® 2008 Team Foundation Server

Pair courses in one location to create a customized training week and save up to $300. See page 4 for more information.

For the latest information on this course and to download a PDF brochure, visit: www.sqetraining.com/vts

2 Days Topical Outline:

Overview
Team System overview
Development challenges and features
Role-based approach
Packaging
LAB

Introducing Visual Studio® 2008
Brief history/overview
Introducing the user interface
Developing solutions
LAB

Team Projects and Process
How process is integrated with Visual Studio® Team System 2008
Overview of MSF for Agile Software Development and for CMMI® Process Improvement

Project Management and Work Teams
Work item tracking system and and handling work items
Creating custom work item queries
Work item integration with Microsoft Project and Microsoft Excel
LAB

Version Control
Architecture overview
Integrated and atomic check-in
Parallel and remote development
Check-in notes, notification, and policy
LAB

Unit Tests
Test-driven development, unit testing, code coverage
LAB

Quality Assurance and Testing
Testing, test manager, writing manual tests, executing manual tests, reporting defects
LAB

Web Testing
Creating and executing Web application and Web services tests
Using extraction and validation rules
Test run configurations and data-driven Web tests
LAB

Load Testing
Creating and executing Web application and service load tests
Using the load agent
Analyzing results
LAB

Build Process
Analyzing test results, working with build reports

Looking to the Future
A look at how the tools will evolve with the next release of Visual Studio® Team System

Experience the Benefits of Self-Paced eLearning

eMASTERING TEST DESIGN

Mastering Test Design: The Art and Science of Creating Test Cases Available on the Web in a Dynamic eLearning Format

Classroom Value with the Convenience of Self-paced Instruction

- Instructed and designed by two of our most experienced instructors, Lee Copeland and Rex Black
- Same valuable information as the two-day classroom course

The Perfect Solution for Test Practitioners with Travel and Time Constraints

- Complete an eLearning course from your own desktop
- Expert mentors provide answers to your specific questions
- You have seventy days to complete the course

What to Expect

- Superior lesson content developed and delivered by testing experts
- Tutorials that place content into real world situations
- Exercises that immediately apply your new learning
- Assessment questions that help you evaluate your learning
- Questions linked to content to reinforce your learning
- Video and audio clips to enhance your learning experience
- Web access to an extensive list of additional resources
- Hyperlinks to a glossary of terms used in the course

To see the two-day classroom course information, see page 9.

Dynamic Learning Features:

Audio
Narration by course author with accompanying transcript

Video Clips from the Classroom
Live examples of the content being taught in the classroom setting

Reinforced Learning
Questions linked to content that helps reinforce what you have just learned

Flash Animation
Illustrates and explains content

Course Outline:

Take a free demo today! Visit www.sqetraining.com/eLearning for more information.

RESOURCE CENTER

Conferences

STAR*EAST* and STAR*WEST* (Software Testing Analysis & Review conferences)
A gathering place for software testers, developers, and managers, these premier software testing events promote interaction on improving software testing practices. STAR's unique, real-world approach delivers the latest testing advances and strategies being used by leading software organizations. The five-day STAR conferences feature international testing experts in keynote sessions, concurrent sessions on testing related topics, in-depth tutorials; and the Testing EXPO with the latest testing tools and services. Visit **www.sqe.com/stareast** and **www.sqe.com/starwest** for more information.

Better Software Conference & EXPO
The Better Software Conference & EXPO delivers the latest in agile and plan-driven software development practices, technology, and solution providers. Exploring improvement throughout the software development lifecycle, the Better Software Conference & EXPO gives you the information you need to be more successful in your software projects by utilizing the latest techniques and technology. Visit **www.sqe.com/bettersoftwareconf** for more information.

Agile Development Practices
The Agile Development Practices conference is for software professionals investigating or implementing agile development practices, processes, technologies, and leadership principles. Explore the latest trends in agile development approaches through keynotes, pre-conference tutorials, EXPO, and concurrent classes in this four-day event. **www.sqe.com/agiledevpractices**

Better Software Magazine

Better Software. It's project management, measurement and metrics, design and architecture, test and evaluation. It's agile methods, plan-driven and requirements-driven processes, software process improvement, business value, and ROI, as well as coverage of emerging technologies. It's special reports, salary surveys, and a steady stream of ideas for software professionals focused on quality. To subscribe, visit **www.BetterSoftware.com**

StickyMinds.com

The most comprehensive online resource and companion to *Better Software* magazine, StickyMinds.com offers an unrivaled scope of original articles from industry experts, technical papers, industry news, Podcasts, white papers, searchable tools and books guides, discussion forums, and more. Membership is free. **www.StickyMinds.com**

Enhance your StickyMinds.com access with a PowerPass membership. Gain premium access to the complete *Better Software* and *STQE* magazine archive and receive the conference materials from every major Software Quality Engineering event, online reference books, and salary surveys, as well as discounts on all Software Quality Engineering products and services. To join, visit **www.StickyMinds.com/PowerPass.asp**

StickyMinds.com publishes FREE eNewsletters delivering new, handpicked content straight to your inbox. Twice a month, the *StickyLetter* brings you great articles, the opinionated "Our Take" column, and lots more. The *What's New Gram* sends you a listing of all the new articles, templates, and book reviews added to StickyMinds.com each week. From the news desk to the desktop, *Between the Lines* brings you industry views of the recent news once a month. The *Sticky ToolLook* delivers a monthly interview with a testing and automation expert. *Fresh Ink* gives subscribers an advance look at what's inside this month's issue of *Better Software* magazine. *iterations* delivers all things agile straight to your inbox. To sign up, visit **www.StickyMinds.com/eLetters.asp**

Reference Books

Software Quality Engineering has published acclaimed reference books addressing the most sought-after topics in software testing. For more information on our books, visit **www.sqe.com/books.asp**

EASY TO REGISTER

Online:
www.sqetraining.com/register

Phone:
888.268.8770
904.278.0524

Email:
sqeinfo@sqe.com

PUBLIC TRAINING COURSES

3 Days — $1,995

Systematic Software Testing

Software Testing Certification—Foundation Level*
There is an additional $250 fee for the ISTQB™ exam.

Writing Testable Requirements

Just-in-Time Software Testing

Test Management

2 Days — $1,495

Mastering Test Design

Lean-Agile Testing Practices

Software Security Testing and Quality Assurance

Requirements-Based Testing

Performance, Load, and Stress Testing

Exploratory Testing in Practice

Testing Applications with Visual Studio® Team System 2008

Visual Studio® 2008 Team Foundation Server

1 Day — $795

Test Process Improvement

eLEARNING COURSES

eMastering Test Design — $695

Self-paced course. Students are allowed seventy days to complete the course.

**** For more information on ISTQB™ certification or to download the syllabus, please visit www.astqb.org.***

BUILD-YOUR-OWN TRAINING WEEK

Build a full week of training in the same location and save up to $300.

Maximize the impact of your training by combining courses in one location to create a customized training week. Pair two courses and save up to $300. See page 4 for more details. For a complete list of courses available, visit **www.sqetraining.com** or call 888.268.8770 or 904.278.0524 for pairing discount options.

ON-SITE TRAINING

For more information about on-site training courses, contact SQE Training at 904.278.0524 or 888.268.8770 or email onsitetraining@sqe.com.

Training Course Fee Includes**

- Tuition
- Course notebook
- Reference book (if applicable) *Textbooks are distributed for public courses only.*
- Refreshment breaks
- Breakfasts and lunches
- Certificate of completion

***$39 of your registration fee includes a one-year digital subscription to* Better Software *magazine. If you are a current subscriber, your subscription will be extended an additional ten issues.*

Training Course Schedule

7:30 a.m. - 8:30 a.m.	Registration (on first day of course) and continental breakfast
8:30 a.m. - 12:00 p.m.	Course
12:00 p.m. - 1:00 p.m.	Lunch
1:00 p.m. - 5:00 p.m.	Course

Satisfaction Guarantee: SQE Training is proud to offer a 100% satisfaction guarantee. We are committed to providing you with the highest quality education and training products. If we are unable to satisfy you, we will gladly refund your registration fee in full.

Public Training Policies: SQE Training reserves the right to make changes in course schedules, dates, locations, and accommodations. We will make every effort to notify students within a reasonable period of time. However, SQE Training is not responsible for personal travel, accommodations, or other incidental expenses in connection with changes to a course.

Cancellation Policy: Attendee substitutions are permitted. Registrants who fail to attend are subject to the full fee if they have not obtained a cancellation code from SQE Training at least six business days prior to the event start date. To obtain a cancellation code, call 904.278.0524 or 888.268.8770.

Register Early: The number of students per course is limited, and many courses fill to capacity. Register early to ensure your space in your preferred course.

Forms of Payment Accepted:

- Visa, MasterCard, or American Express
- Check or company purchase order is accepted. However, payment must be received before course registration is confirmed.

Confirmation: After payment, you will receive a confirmation notice containing course details (e.g., hotel, accommodations). Please bring the letter to the course for admittance.

Build-Your-Own Testing Training Week

Pair two courses in the same location and save up to $300!

See page 4 for details.

Accelerate Your Career & Empower Your Team

SOFTWARE TESTING TRAINING

NEW FALL 2008 SCHEDULE

TEST ENGINEERING

- Lean-Agile Testing Practices
- Systematic Software Testing
- Mastering Test Design
- Software Testing Certification
- Just-in-Time Software Testing
- Performance, Load, and Stress Testing

TEST MANAGEMENT

- Test Management
- Test Process Improvement
- Visual Studio Team System Training
- ... and Many More Courses Inside

SQE TRAINING
www.sqetraining.com

- Public Training
- On-site Training
- eLearning
- Consulting

R.E.P. Project Management Institute

ISTQB

EASY TO REGISTER

Online: www.sqetraining.com/register

Phone: 888.268.8770 904.278.0524

Email: sqeinfo@sqe.com

98% of our students in 2007 said their training experiences met or exceeded their expectations.

SQE Training
330 Corporate Way, Suite 300
Orange Park, FL 32073

IF ADDRESSEE IS NO LONGER EMPLOYED:
Re-route to Director of Software Development