

QAI - Certified Associate in Software Testing

Overview

As the IT industry becomes more competitive, the ability for management to distinguish professional and skilled individuals in the field becomes mandatory. CAST Certification demonstrates a foundation level understanding of quality testing principles and practices.

Acquiring the designation of Certified Associate in Software Testing (CAST) indicates a professional level of competence in the principles and practices of software testing in the IT profession. CASTs become members of an acclaimed professional group, receiving recognition of their competence by business and professional associates, and are afforded potentially more rapid career advancement.

The Certified Associate In Software Testing Preparatory Course has been designed specifically to prepare the software quality testing professional for the CAST examination.

Who Should Attend?

- Testers of all types and levels
- Other disciplines who perform their own testing or are involved in testing
- Quality Assurance Professionals
- Test Management
- QA Managers
- QA Directors
- Software Engineers
- Business Analysts
- Project Managers
- IT Specialists (Security, Capacity Management, Networking...)
- Business Stakeholders
- Outsourcer Staff (Buyers and Suppliers)
- Application Development Managers

Course Outcome

The Certified Associate in Software Testing program is intended to establish standards for initial qualification and continuing improvement of professional competence. This certification program helps to:

- Define the tasks (skill categories) associated with software testing duties in order to evaluate skill mastery.
- Demonstrate an individual's willingness to improve professionally.
- Acknowledge attainment of an acceptable standard of professional competency.
- Aid organizations in selecting and promoting qualified individuals.
- Motivate personnel having software testing responsibilities to maintain their professional competency.
- Assist individuals in improving and enhancing their organization's software testing programs.

Outcome

After the completion of the course, the participants would be able to:

- Understanding of the basic concepts of Software testing and the various types of testing and the necessity of Test architecting
- Understand the behavioral aspects of a tester (Think like a tester)
- Understand the Role of Test Architect and steps involved in the management of software life cycle
- Understand the Principles of Test architecting
- Understand and apply the rudimentary rules of test architecting and managing test activities
- Understand the principles of Test Automation frameworks and Performance Engineering

Duration

5 days

Certificate

Certified Associate in Software Testing

Certification Body

Course Outline

The content of this course is designed to support the course objectives.

1. Software Testing Principles and Concepts - This skill category focuses on the “basics” of software testing as represented by the vocabulary of testing, testing approaches, methods and techniques as well as the materials used by testers in performing their test activities.
2. Building the Test Environment - The test environment is comprised of all the conditions, circumstances, and influences surrounding and affecting the testing of software. This category’s discussions include assuring the test environment fairly represents the production environment to enable realistic testing to occur.
3. Managing the Test Project - Software testing is a project with almost all the same attributes as a software development project. This category discusses project planning, project staffing, scheduling and budgeting, communicating, assigning and monitoring work and ensuring that changes to the project plan are incorporated into the test plan.
4. Test Planning - This skill category explores the skills testers need to plan tests, including the selection of techniques and methods to be used to validate the product against its approved requirements and design. Test planning assesses the software application risks, and then develops a plan to determine if the software minimizes those risks. Testers must understand the development methods and environment to effectively plan for testing.
5. Executing the Test Plan - The test plan should be executed as designed. If the plan cannot be executed as designed it should be changed, or notations made as to what aspects of the plan were not performed. Testing according to the test plan should commence when the project commences and conclude when the software is no longer in operation. Portions of the test plan can be performed while the test plan is being written. This category focuses on the many skills needed to carry out the test plan, like design test cases and test scripts, use test tools, execute tests, record test results, and manage defects.
6. Test Status, Analysis and Reporting - In this category, the tester’s ability to develop testing status reports is discussed. These reports should show the status of the testing based on the test plan. Reporting should document what tests have

been performed and the status of those tests. The test reporting process is a process to collect data, analyze the data, supplement the data with metrics, graphs and charts and other pictorial representations which help the developers and users interpret that data. The lessons learned from the test effort should be used to improve the next iteration of the test process.

7. User Acceptance Testing - As much as possible, once the acceptance criterion has been established, they should be integrated into all aspects of development. This category discusses how the same process can be used by software testers when users are unavailable for test; when diverse users use the same software; and for beta testing software. Although acceptance testing is a customer and user responsibility, testers normally help develop an acceptance test plan, include that plan in the system test plan to avoid test duplication; and, in many cases, perform or assist in performing the acceptance test.
8. Testing Software Developed by Outside Organizations - This skill category reviews the many challenges faced when testing software developed by a contractor, or an external organization. It is management's responsibility that acquired software meets the needs of their organization. Contractors will test the software they build, but that does not relieve management from their quality responsibilities.
9. Testing Software Controls and the Adequacy of Security Procedures - This category's discussions focus on testing internal control as it is a key issue for software testers. Security is a component of internal control that warrants special attention of testers. Interest in internal control has been highlighted by publicized penetrations of security and the increased importance of information systems and the data contained by those systems.
10. Testing New Technologies - Testers require skills in their organization's current technology, as well as a general understanding of the new information technology that might be acquired by their organization. The new technology skills are required because the test plan needs to be based on the types of technology used. This skill category addresses why new IT technology, as well as any technology new to the testers or the organization must be addressed in the test plan.
11. Mock Tests