COMPUTER & TECHNOLOGY PROGRAMS

Test Planning, Modeling and Reporting

Take a structured approach to test case development as you plan recurring test activities and examine the product fundamentals necessary for decision making. Recognize tester's contribution to the product team and their role in an agile environment. Students will develop advanced skills in test planning, prioritization and modeling. As part of the curriculum, students will gain new skills and insight into test effectiveness, the measurement of test progress and identification of risks associated with complex software development projects. Topics: test plan writing, test verification and validation, project metrics, user acceptance, testing to spec and common enterprise environments.

Who should take this course?

This course is designed for students who are focusing on software testing as a career. Students who want to develop advanced techniques at risk assessment and effective skills at measuring software quality should take this course. Students looking to gain advanced knowledge and perspective in the role of testing in the Software Development Lifecycle are highly encouraged to take this course. It is expected that the student is already familiar with C# and software testing. This course is a requirement for the Software Testing Certificate.

Course Objectives

- Use industry standard models to review and incorporate testing techniques and test case development for large features or the entire product.
- Write and run test cases that help to establish a baseline of quality for a feature during development, or from one version of the product to the next.
- Enumerate issues and scenarios which can affect the priority of testing, severity of issues found, various customers, and/or test resources needed for product testing.
- Identify how testing is created, run and prioritized, factoring in collaboration with team members.
- Enumerate the components of the product under test, accounting for interactions between subcomponents, code used or reused, project schedule and testing resources.
- Develop and present quality metrics which are commonly agreed upon, measurable, maintainable, and provide the minimum necessary data for the intended audience(s).

Course Details

- Length: 21 hours
- Format: Classroom
- Prerequisites: Principles of Software Testing or equivalent
  The above prerequisites are considered to be the basic skills and knowledge needed prior to taking this class. Instructors will assume your readiness for the class materials and will NOT use class time to discuss prerequisite materials.

COMPUTER & TECHNOLOGY PROGRAMS

Course Contents

Use industry standard models to review and incorporate testing techniques and test case development for large features or the entire product.

Write and run test cases that help to establish a baseline of quality for a feature during development, or from one version of the product to the next.

- Develop a plan to test a feature at the milestone or project-signoff level using a comprehensive test suite.
- Explain how to plan the test automation counterpart for testing to specification.
- Prioritize test cases based on customer scenarios, importance of the functionality, and product stability for the next stage of development.
- Execute the full test plan to validate all feature functionality of the product during each development milestone.
- Explain how to write a UAT plan, including validation tests that allow clients to verify and comment on feature functionality before the development cycle concludes.
- Describe test activities which happen as part of release or go-live.
- Identify common issues, fixes and user feedback after a product has been released.

Enumerate issues and scenarios which can affect the priority of testing, severity of issues found, various customers, and/or test resources needed for product testing.

- Identify the customer's business goals and the software's value to the customer.
- Enumerate the issues and scenarios related to privacy and regulatory compliance.
- Enumerate the issues and scenarios related to security.
- Enumerate the issues and scenarios related to testing an enterprise scenario.
- Enumerate the issues and scenarios for application and hardware compatibility.
- Enumerate the issues and scenarios for localization, globalization, international, geopolitical issues.
- Enumerate the issues and scenarios for usability, beta testing, user feedback.
- Enumerate the issues and scenarios for client/server architectures.
- Enumerate the issues and scenarios installation, setup, client configuration and first run.
- Enumerate the issues and scenarios for performance, reliability and stress testing.

Course Contents

Identify how testing is created, run and prioritized, factoring in collaboration with team members.

- Identify the tester's role in an Agile environment.
- Identify the tester's role in a classic waterfall schedule.
- Bring a QA perspective to similar or parallel organizations and roles.
- Identify the tester's role at each stage of product development.
- Recognize how software testers contribute to the Product Development Team.
- Identify the tester's role in relation to updates, point releases, and testing after the release of a major version.
- Identify the tester's role in other development methodologies.

Enumerate the components of the product under test, accounting for interactions between subcomponents, code used or reused, project schedule and testing resources.

- Write a test plan that gives testers the basis to write test cases for the feature under test and draws on available documentation.
- Identify the aspects of an effective code review or specification review.
- Describe how to collaborate with developers and automation engineers to develop a comprehensive approach to testing that includes automation.

Develop and present quality metrics which are commonly agreed upon, measurable, maintainable, and provide the minimum necessary data for the intended audience(s).

- Identify the differences between test data that is relevant to testers, vs. aggregated or interpreted data that is relevant to team members, project managers, executives and/or customers.
- Develop metrics which establish and communicate a baseline of quality for the product and produce results that have clear next steps.