

Principles of Software Testing

This course introduces you to the field of software testing through discussion of the tester's role in the product cycle and hands-on labs that give you an opportunity to develop software testing skills. This course covers how testers find and break down software problems, identify bugs, determine types of bugs, report bugs, and perform regression testing. This hands-on class gives you an opportunity to understand the tasks testers execute and then try them yourself.

Who should take this course?

This course is designed for students in multiple disciplines who want to develop basic testing skills which can be applied to the field of software testing, project management and development. Students will learn the fundamentals of testing and structured systems validation. It is expected that the student is already familiar with Windows and basic Internet technologies. This course is a prerequisite for taking more advanced courses in Software Testing.

Course Objectives

- Describe the tester's role in software development, the skills expected of a tester, and areas of specialization within the field.
- Evaluate a subsection of software at the bug or feature level, by searching for the most serious or common issues.
- Enumerate common classes of bugs according to why the bugs may be a problem.
- Write an issue report that provides the minimal actionable data necessary to reproduce a bug, prioritizes an issue correctly, and utilizes the terminology and metadata of the project's bug tracking system.
- Perform regression testing to ensure the original issue is fixed and new issues of equal or greater severity have not been introduced into the code.
- Write a block of test cases to validate bugs are fixed and to find additional bugs.

Course Details

- Length: 21 hours
- Format: Classroom
- Prerequisites: Familiarity with Windows
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.



Course Contents

Describe the tester's role in software development, the skills expected of a tester, and areas of specialization within the field.

- Summarize the techniques testers use to break down problems, organize their testing and report their findings, at the issue or feature level.
- Summarize the stages of the product cycle that involve testers.
- Summarize common areas in which testers specialize.
- Summarize the systematic approaches testers use to break down problems, organize their testing and report their findings, at the feature or product level.

Evaluate a subsection of software at the bug or feature level, by searching for the most serious or common issues.

- Describe basic strategies to test any code, feature, software or software environment.
- Describe the terms, processes, and programs that will help you navigate the test environment within the software development organization.

Enumerate common classes of bugs according to why the bugs may be a problem.

- Identify and classify common classes such as coding errors, design issues, customer requirements issues, documentation errors, functional specification errors, feature or enhancement requests, and environmental conditions.
- Identify the common sections of a test case that enable a tester to learn about a test area, find and report bugs, and determine when the test case is outdated or incomplete.

Course Contents

Write bug reports that provide the minimal actionable data necessary to reproduce a bug, prioritizes an issue correctly, and utilizes the terminology and metadata of the project's bug tracking system.

- Describe the bug reporting lifecycle as it relates to software development lifecycle.
- Write a bug report that clearly describes the minimal steps to reproduce a software defect.
- Prioritize an issue against other expected behaviors, taking into account the stage of product development, importance of the feature / functionality, and when known, complexity of the fix or dependent features.
- Write a bug using the terminology, and minimum metadata of the project's bug tracking system(s).

Perform regression testing to ensure the original issue is fixed and new issues of equal or greater severity have not been introduced into the code.

- List the major ways in which software fixes are delivered. Explain how to test a fix early in the software development life cycle.
- Create regression test suites to validate bugs are fixed and to find additional bugs at project milestones.

Write a block of test cases to validate bugs are fixed and to find additional bugs.

- Describe the components of a test case and how to make it effective and relevant to its environment.