ProTech Professional Technical Services, Inc.

Course Outline

# Effective Methods of Software Testing

## Course Summary

### Description

Software testing comprises a wide range of practices, techniques and disciplines integrated under a common body of professional knowledge, protocols and standards.  This three day course is a comprehensive overview of the various dimensions of software testing presented within the context of the established theory and professional protocols universally used by software testers.

The primary objective of the course is to provide students with a solid foundation in the concepts and professional practices of software testing rather than focusing on the details of any specific tool or technique.  Once this professional foundation is established, students can then both move into more advanced and specialized topics, and perform critical analyses of their own testing practices and techniques.

The course also covers the fundamentals of test planning, execution and the professional standards governing the test life cycle from establishing testing objectives, to test design and test execution and the reporting and analysis of the test results.

### Audience

This course is designed to either be taken as an introductory course in preparation for more advanced testing courses, or as a stand alone course for those who need an comprehensive understanding of software testing but will not be working in the field or taking further courses.

### Topics

- Introduction to Software Testing
- Good Testing. Bad Testing
- Testing Protocols and Concepts
- Types of Testing
- Functional Testing
- Structural Testing
- Test Management
- Non-Functional Testing
- Reviews and Inspections
- Logic and Model Testing
- Agile Testing
- The Testing Maturity Model
- Specialized Testing
- Test Improvement
- Conclusion

### Prerequisite

There are no prerequisites for the course.

### Duration

Three Days

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names.  References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies.  It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically.

# Effective Methods of Software Testing

## Course Outline

### I. *Introduction to Software Testing*

- A. Why we test and the value of testing
- B. The evolution of software testing
- C. The Beizer levels of a tester’s development
- D. The current state of software testing
- E. Software testing as a special case of general testing

### II. *Good Testing. Bad Testing*

- A. Defining effective versus ineffective testing
- B. Costs of ineffective testing
- C. Test validity, accuracy, reliability and comprehensiveness
- D. Preventing errors in test planning
- E. Preventing errors in test design
- F. Preventing errors in test execution
- G. Sources of human error
- H. IEEE best practices checklist

### III. *Testing Protocols and Concepts*

- A. The principles of testing
- B. Errors, faults and failures
- C. Testing protocols: how they work and why we use them
- D. Testing and quality control and quality management
- E. Testing and quality assurance
- F. Testing throughout the development lifecycle
- G. Automated testing
- H. The professional testing mindset
- I. Professional standards and ethics

### IV. *Types of Testing*

- A. Test levels: unit, component, integration, system and acceptance
- B. Confirmatory versus evaluative testing
- C. Unit testing and testing simplification
- D. Robustness and correctness testing
- E. Integration testing methods
- F. Structural versus functional versus performance testing
- G. Acceptance testing methods
- H. Static testing: reviews and inspections
- I. Other types of testing

### V. *Functional Testing*

- A. The black box methodology
- B. Equivalence class analysis
- C. Boundary value analysis
- D. Cause and effect analysis
- E. Algorithmic test case generation
- F. Testing the specification
- G. Coverage and sensitivity measures
- H. What functional testing can and cannot do

### VI. *Structural Testing*

- A. The white box methodology
- B. Flow of control analysis
- C. Coverage levels
- D. Structural unit testing
- E. Data flow testing
- F. Structural testing of business logic, use cases and user stories

### VII. *Test Management*

- A. Developing quality objectives and test goals
- B. Test management standards and best practices
- C. Reusable test models
- D. Writing the test plan
- E. Managing test designs and execution protocols
- F. Test case management
- G. Managing the testing activities
- H. Test process metrics and process improvement

### VIII. *Non-Functional Testing*

- A. Performance and "-ility" types of testing
- B. Unique challenges in non-functional testing
- C. Test design for non-functional tests
- D. Basic concepts of test engineering
- E. Planning non-functional testing

### IX. *Reviews and Inspections*

- A. The value of reviews and inspections
- B. The role of the tester in reviews and inspections
- C. How to conduct an effective review
- D. The role of reviews in quality assurance

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically.

## Effective Methods of Software Testing

### Course Outline(cont.)

X. *Logic and Model Testing*

A. Understanding the role of models in development<br>
B. Model types: domain, design, business, data, etc<br>
C. Poor models as sources of error<br>
D. Testing and evaluating models structurally<br>
E. Testing and evaluating logic<br>
F. When and where to test models<br>
G. Integrating model testing into the test process

XI. *Agile Testing*

A. Overview of adaptive and Agile/DevOps processes<br>
B. Adapting testing activities to an Agile process<br>
C. The Agile testing quadrants<br>
D. Confirmatory testing – Test Driven Development<br>
E. Confirmatory testing – Acceptance Test Driven Development<br>
F. Scenario based testing<br>
G. Exploratory testing

XII. *The Testing Maturity Model*

A. Process maturity concepts and the CMM model<br>
B. The Test Maturity Model<br>
C. How a testing organization matures<br>
D. Dependencies between process maturity and test maturity<br>
E. Identifying anti-maturity practices

XIII. *Specialized Testing*

A. Security testing<br>
B. Usability testing<br>
C. Environmental testing<br>
D. Risk based testing<br>
E. DevOps and continuous testing

XIV. *Test Improvement*

A. Test improvement and process improvement<br>
B. Managing testing with metrics<br>
C. Evaluating the effectiveness of testing<br>
D. Developing a test process improvement process<br>
E. Test improvement and quality management

XV. *Conclusion*

A. Future trends in testing and quality<br>
B. Adapting to emerging technologies, eg. testing AI applications<br>
C. Wrap up and summary<br>
D. Other topics as requested

Due to the nature of this material, this document refers to numerous hardware and software products by their trade names. References to other companies and their products are for informational purposes only, and all trademarks are the properties of their respective companies. It is not the intent of ProTech Professional Technical Services, Inc. to use any of these names generically.