IT-2200: SOFTWARE QUALITY ASSURANCE TECHNIQUES

Cuyahoga Community College

Viewing: IT-2200 : Software Quality Assurance Techniques

Board of Trustees:

November 2020

Academic Term:

Fall 2021

Subject Code

IT - Information Technology

Course Number:

2200

Title:

Software Quality Assurance Techniques

Catalog Description:

Continuation of Software Quality Assurance process covering testing types, techniques and test management cycle and will also get exposure to Agile Testing. Gain practical experience creating and executing test cases and plans, logging and tracking defects etc.

Credit Hour(s):

4

Lecture Hour(s):

3

Lab Hour(s):

2

Requisites

Prerequisite and Corequisite

IT-1200 Software Quality Assurance.

Outcomes

Course Outcome(s):

Use knowledge of how testing integrates with software development team.

Objective(s):

1. Explain Software Development Life Cycle.
2. Describe Software Development Lifecycle Models – Sequential development model and Iterative and incremental model - Rational Unified Process, Scrum, Kanban, Spiral, and Agile Development.
3. Identify the benefits and challenges for testers in iterative and incremental SDLC environments.

Course Outcome(s):

Define and apply various test design methodologies to appropriate testing scenarios.

Essential Learning Outcome Mapping:

Critical/Creative Thinking: Analyze, evaluate, and synthesize information in order to consider problems/ideas and transform them in innovative or imaginative ways.

Objective(s):

1. Explain the static techniques and the test process.
2. Identify the work products that can be examined by static testing.
3. Explain the difference between static and dynamic testing.
4. Explain the work product review process.

5. Describe the roles and responsibilities in a formal review.
6. Explain the types of reviews – Technical review & Inspection.
7. Describe how to apply review techniques – Ad hoc reviewing, Checklist based reviewing, Scenario based reviewing and Role-based reviewing, Perspective based reading.
8. Explain categories of test techniques and characteristics.
9. Explain Equivalent partnering.
10. Explain Boundary value analysis.
11. Describe how to use decision tables for test design.
12. Explain state transition testing.
13. Describe Usecase testing.
14. Explain test coverage.
15. Describe types of coverage.
16. Explain how to measure coverage.
17. Define Statement testing coverage.
18. Define error guessing technique.
19. Explain exploratory testing technique.
20. Describe checklist-based testing technique.

Course Outcome(s):

Develop test plans using the scientific method that meets user acceptance criteria based on existing code and allows plans to be repeatable (i.e. performance, user acceptance, regression).

Objective(s):

1. Describe Independent testing.
2. Identify levels of independent testing.
3. Explain the potential benefits and drawbacks of test independence.
4. Identify the tasks of a test manager and a tester.
5. Explain the test manager tasks.
6. Describe the tester tasks.
7. Explain the purpose and content of test plan.
8. Describe test strategy and test approach.
9. Define entry criteria and exit criteria (Definition of Ready and Done).
10. Describe how to create a test execution schedule.
11. Identify the factors influencing test effort.
12. Describe how to estimate what testing will involve and what it will cost.
13. Identify the factors that affect the test effort.
14. Explain test estimation techniques.
15. Define Entry criteria and Exit criteria (Definition of Ready and Done).
16. Explain how to choose a test strategy.
17. Describe how to create a test execution schedule.
18. Identify the factors influencing test effort.
19. Explain test estimation - what testing will involve and what it will cost.
20. Identify the factors that affect the test effort.
21. Explain the various test estimation techniques and their pros/cons.
22. Define risk.
23. Explain product and project risk.
24. Explain Risk-based testing and product quality.

Course Outcome(s):

Perform testing on software including API/Web service, Web, Desktop, and Mobile. (Response Web Design)

Objective(s):

1. Describe Test tool classification.
2. Describe Tool support for management of testing and software.
3. Describe Tool support for static testing.
4. Describe Tool support for test execution and logging.

5. Describe Tool support for performance measurement and dynamic analysis.
6. Describe Tool support for specialized testing needs.
7. Describe the benefits and risk of test automation.
8. Identify the benefits of using tools.
9. Identify the risks of using tools.
10. Identify Test execution tools.
11. Identify Test Management tools.
12. Explain the principles for Tool selection.
13. Describe the success factors for Tools.
14. Identify the tools that can be used to test front end web pages, backend code (headless), APIs, and Databases.
15. Identify the limitations of these tools.

Course Outcome(s):

Create documentation of system defects with sufficient detail and advocating for correction to meet customer needs.

Objective(s):

1. Explain the purpose of metrics in testing.
2. Identify the common test metrics.
3. Define the purpose of test reports.
4. Identify the content of test reports.
5. Identify the audience of test reports.
6. Describe how to uniquely identify and track all testing workproducts and defect reports.
7. Explain the objectives of a defect report.
8. Describe how to write a good defect report.
9. Identify what goes in a defect report.
10. Explain what happens to defect reports after you file them.

Methods of Evaluation:

1. Quizzes
2. Discussion Board
3. Labs
4. Final Project
5. Exams

Course Content Outline:

1. Fundamentals of testing
    a. What is testing?
    b. Objectives of testing
    c. Testing and Debugging
    d. Why is testing necessary?
        i. Testing contributions to success
        ii. Errors defects and failures
        iii. Defects root causes and effects
    e. Seven testing principles
2. Test Process
    a. Test activities and tasks
    b. Test planning
    c. Test analysis
    d. Test Implementation
    e. Test Completion
3. Test work products
    a. Test planning work products
    b. Test Monitoring and control work products
    c. Test Design work products
    d. Test implementation work products
    e. Test execution work products

    f. Test completion work products\
    g. Traceability between test basis and test work products
4. Psychology of testing
    a. Human psychology and testing
        i. Finding defects
        ii. Bia
        iii. Review and test your own work?
        iv. Attitude
        v. Communication
    b. Mindset of testers and developers
5. Testing throughout the Software Development Lifecycle
    a. Software Development Lifecycle Models - Sequential development model, Iterative and incremental model
    b. Testing in iterative and incremental model, Rational Unified Process, Scrum, Kanban, Spiral, Agile Development
6. Test Levels
    a. Component testing
    b. Integration testing
    c. System testing
    d. Acceptance testing
7. Test Types
    a. Functional testing
    b. Non-functional testing
    c. Whitebox testing
    d. Change related testing
8. Maintenance testing
    a. Triggers for maintenance
    b. Impact analysis and regression testing
9. Static techniques
    a. Static techniques and the test process
    b. Work products that can be examined by static testing
    c. Diff between static and dynamic testing
10. Review process
    a. Work product review process
    b. Roles and Responsibilities in a formal review
    c. Types of review – Technical review, Inspection
    d. Applying review techniques
        i. Ad hoc reviewing
        ii. Checklist based reviewing
        iii. Scenario based reviewing
        iv. Role-based reviewing
        v. Perspective based reading
    e. Success factors for Reviews
11. Test Techniques
    a. Categories of test techniques
    b. Choosing a test technique
    c. Categories of test techniques and characteristics
12. Blackbox techniques
    a. Equivalent partnering
    b. Boundary value analysis
    c. Extending equivalent partnering and boundary value analysis
    d. Using design tables for test design\
    e. State transition testing
    f. Testing for invalid transition
    g. Usecase testing
13. Whitebox techniques
    a. What is coverage?
    b. Types of coverage
    c. How to measure coverage
    d. Statement testing and coverage

    e. Decision testing and coverage
    f. The value of statement and decision testing

14. Experience based techniques
    a. Error guessing
    b. Exploratory testing
    c. Checklist based testing
15. Test Management
    a. Test organization
    b. Test planning and estimation
    c. Test monitoring and control
    d. Defect management
    e. Configuration Management
16. Test Organization
    a. Independent testing
    b. Levels of independent testing
    c. Potential benefits and drawbacks of test independence
17. Tasks of a test manager and a tester
    a. Test manager tasks
    b. Tester tasks
    c. Defining the skills test staff needs
18. Test Planning and Estimation
    a. The purpose and content of testplan
    b. What do you do with your brain when planning tests
19. Test strategy and test approach
    a. Entry criteria and Exit criteria (Definition of Ready and Done)
    b. Test execution schedule
    c. Factors influencing test effort
    d. Estimating what testing will involve and what it will cost
    e. Factors that affect the test effort
    f. Test estimation techniques
20. Test monitoring and control
    a. Metrics used in testing
    b. The purpose of metrics in testing
    c. Common test metrics
21. Purpose, Contents and Audience for test report
    a. Purpose of test reports
    b. Content of test reports
    c. Audience of test reports and the effect on the report
22. Risks in testing
    a. The definition of risk
    b. Product and project risk
    c. Risk-based testing and product quality
23. Defect Management
    a. Objectives of a defect report
    b. How to write a good defect report
    c. What goes in a defect report
    d. What happens to defect reports after you file them
24. Tool support for testing
    a. Test tool considerations
    b. Test tool classification
    c. Tool support for management of testing and software
        i. Tool support for static testing
        ii. Tool support for test execution and logging
        iii. Tool support for performance measurement and dynamic analysis
        iv. Tool support for specialized testing needs
        v. Benefits and risk of test automation
        vi. Benefits of using tools
        vii. Risks of using tools

   viii. Test execution tools
   ix. Test Management tools

25. Effective use of tools
    a. Principles for Tool selection
    b. Success factors for Tools
26. Tools for testing Web/API
    a. List of tools that can be used for testing and their limitation.
    b. front end web pages
    c. backend code (headless)
    d. APIs
    e. Databases

Resources

Dorothy Graham, Erik P. W. M. Veenendaal, Rex Black. *Foundations of Software Testing ISTQB Certification*. 4th edition. Cengage, 2019.

Paul C. Jorgensen. *Software Testing: A Craftsman's Approach*. Fourth ed. Boca Raton: CRC Press, 2014.

Brian Hambling , Peter Morgan , Angelina Samaroo, Geoff Thompson, Peter Williams. *Software Testing: An ISTQB-BCS Certified Tester Foundation Guide*. Fourth Edition. United Kingdom: BCS Learning and Development, LTD, 2019.

Cem Kaner , James Bach. *Lessons Learned in Software Testing: A Context-Driven Approach*. John Wiley & Sons, Inc., 2001.

Gerard O'Regan. *Concise Guide to Software Testing (Undergraduate Topics in Computer Science)*. Switzerland: Springer, 2019.

Top of page

Key: 4863