

MC5503 SOFTWARE TESTING AND QUALITY ASSURANCE

DETAILED SYLLABUS

OBJECTIVES:

- To know the behavior of the testing techniques and to design test cases to detect the errors in the software
- To get insight into the levels of testing in the user environment
- To understand standard principles to check the occurrence of defects and its removal.
- To learn the functionality of automated testing tools to apply in the specialized environment.
- To understand the models and metrics of software quality and reliability.

UNIT I TESTING TECHNIQUES & TEST CASE DESIGN

Using White Box Approach to Test design - Test Adequacy Criteria – Static Testing Vs. Structural Testing – Code Functional Testing – Coverage and Control Flow Graphs – Covering Code Logic – Paths – Their Role in White box Based Test Design – Code Complexity Testing – Evaluating Test Adequacy Criteria. Test Case Design Strategies – Using Black Box Approach to Test Case Design – Random Testing – Requirements based testing – Boundary Value Analysis – Decision tables – Equivalence Class Partitioning – State based testing – Cause-effect graphing – Error guessing – Compatibility testing – User documentation testing – Domain testing – Case study for Control Flow Graph and State based Testing.

UNIT II LEVELS OF TESTING

The Need for Levels of Testing- Unit Test Planning – Designing the Unit Tests – The Test Harness – Running the Unit tests and Recording Results – Integration Tests – Designing Integration Tests – Integration Test Planning – Scenario Testing – Defect Bash Elimination. System Testing – Acceptance testing – Performance testing – Regression Testing - Internationalization testing - Ad-hoc testing – Alpha, Beta Tests- Testing OO systems – Usability and Accessibility Testing – Configuration Testing - Compatibility Testing – Testing the documentation – Website Testing - Case Study for Unit and Integration Testing.

UNIT III TESTING FOR SPECIALIZED ENVIRONMENT

Testing Client / Server Systems – Testing in a Multiplatform Environment - Testing Object Oriented Software – Object Oriented Testing – Testing Web based systems – Web based system – Web Technology Evolution – Traditional Software and Web based Software – Challenges in Testing for Web-based Software – Quality Aspects – Web Engineering – Testing of Web based Systems. Case Study for Web Application Testing

UNIT IV TEST AUTOMATION

Selecting and Installing Software Testing Tools - Software Test Automation – Skills needed for Automation – Scope of Automation – Design and Architecture for Automation – Requirements for a Test Tool – Challenges in Automation – Tracking the Bug – Debugging – Case study using Bug Tracking Tool.

UNIT V SOFTWARE TESTING AND QUALITY METRICS

Six-Sigma – TQM - Complexity Metrics and Models – Quality Management Metrics - Availability Metrics - Defect Removal Effectiveness - FMEA - Quality Function Deployment – Taguchi Quality Loss Function – Cost of Quality. Case Study for Complexity and Object-Oriented Metrics.

REFERENCES:

1. Adithya P. Mathur, “Foundations of Software Testing – Fundamentals algorithms and techniques”, Dorling Kindersley (India) Pvt. Ltd., Pearson Education, 2008
2. Boris Beizer, “Software Testing Techniques”, Dream Tech Press, 2009
3. Dale H. Besterfield, “Total Quality Management”, Pearson Education Asia, Third Edition, Indian Reprint (2011).
4. Edward Kit, “Software Testing in the Real World – Improving the Process”, Pearson Education, 1995
5. Glenford J. Myers, Tom Badgett, Corey Sandler, “The Art of Software Testing”, 3rd Edition, John Wiley & Sons Publication, 2012
6. Illene Burnstein, “Practical Software Testing”, Springer International Edition, Chennai, 2003.
7. Naresh Chauhan, “Software Testing Principles and Practices” Oxford University Press, New Delhi ,2010
8. Ron Patton, “Software Testing”, Second Edition, Pearson Education, 2009