

## SOFTWARE TESTING METHODOLOGY

| <b>VII Semester: CSE / IT</b>   |             |                               |   |   |                          |               |     |       |
|---|-------------|-------------------------------|---|---|--------------------------|---------------|-----|-------|
| Course Code   | Category    | Hours / Week                  |   |   | Credits                  | Maximum Marks |     |       |
|   |             | L                             | T | P |                          | CIA           | SEE | Total |
| AIT008  | <b>Core</b> | 3                             | 1 | - | 4                        | 30            | 70  | 100   |
|   |             | <b>Practical Classes: Nil</b> |   |   | <b>Total Classes: 60</b> |               |     |       |
| <p><b>CONTACT CLASSES: 45    TUTORIAL CLASSES: 15</b></p>   |             |                               |   |   |                          |               |     |       |
| <p><b>OBJECTIVES:</b></p> <p><b>The course should enable the students to:</b></p> <ol style="list-style-type: none"> <li>I. Understand the concept of software testing objectives, process criteria, strategies and methods.</li> <li>II. Demonstrate various software testing issues and solutions in software like unit test, integration, regression and system testing.</li> <li>III. Demonstrate the techniques and skills on how to use modern software testing tools to support software testing projects.</li> <li>IV. Understand important concepts of complexity metrics and object oriented metrics.</li> </ol>  |             |                               |   |   |                          |               |     |       |
| <p><b>COURSE OUTCOMES:</b></p> <ol style="list-style-type: none"> <li>1. Understand the basic concepts of testing, path testing and sensitization</li> <li>2. An Ability to learn about the transaction flow testing</li> <li>3. Understand the concepts of domain based testing and logic based testing</li> <li>4. To describe about the path product and data flow anomaly detection</li> <li>5. Understand the concepts of transitions testing.</li> </ol>  |             |                               |   |   |                          |               |     |       |
| <p><b>COURSE LEARNING OUTCOMES:</b></p> <ol style="list-style-type: none"> <li>1. Explain the importance of testing and purpose of testing.</li> <li>2. Illustrate different dichotomies of testing.</li> <li>3. Demonstrate the model for testing, different testing levels and role of models.</li> <li>4. Describe the consequences and taxonomy of bugs and different bugs in project environment.</li> <li>5. Illustrate the concepts of path testing, predicate loops and path sensitization.</li> <li>6. Explain Path instrumentation and their applications</li> <li>7. List out the Transaction flows techniques, structures and their test databases.</li> <li>8. State the basics of data flow testing, Strategies in data flow testing and applications of dataflow testing</li> <li>9. Describe Domains, paths and explain about bugs and their tools.</li> <li>10. Demonstrate Domains and Interfaces testing.</li> <li>11. Explain about the line arising transformation and coordinate transformation.</li> <li>12. Describe Logic based testing, Decision tables and compare hardware and software testing..</li> <li>13. Illustrate Path expression, KV Charts and their specifications.</li> <li>14. State Path products and path expression, different laws used in path testing.</li> <li>15. Demonstrate the Reduction procedure.</li> <li>16. Explain about the Regular expressions.</li> <li>17. Explain about Flow anomaly detection.</li> <li>18. Explain State Graphs and state testing and their Testability Tips.</li> <li>19. Explain about good and bad state graphs.</li> <li>20. Explain finite state behavior in state graphs.</li> </ol> |             |                               |   |   |                          |               |     |       |

|  |                                 |                    |
|--|---------------------------------|--------------------|
| <b>UNIT-I</b>  | <b>INTRODUCTION TO TESTING</b>  | <b>Classes: 10</b> |
| Introduction: Purpose of testing, dichotomies, model for testing, consequences of bugs, taxonomy of bugs. Flow graphs and path testing: Basics concepts of path testing, predicates, path predicates and Achievable paths, path sensitizing, path instrumentation, application of path testing.  |                                 |                    |
| <b>UNIT-II</b>   | <b>TRANSACTION FLOW TESTING</b> | <b>Classes: 08</b> |
| Transaction flow testing: Transaction flows, transaction flow testing techniques, dataflow testing, basics of dataflow testing, strategies in dataflow testing, application of dataflow testing.   |                                 |                    |
| <b>UNIT-III</b>  | <b>LEVELS OF TESTING</b>        | <b>Classes: 09</b> |
| Domain testing: Domains and paths, nice and ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.<br>Logic based testing: Overview, decision tables, path expressions, kv charts, and specifications.   |                                 |                    |
| <b>UNIT-IV</b>   | <b>PATH PRODUCTS</b>            | <b>Classes: 08</b> |
| Paths, path products and regular expressions: Path products and path expression, reduction procedure, applications, regular expressions and flow anomaly detection.  |                                 |                    |
| <b>UNIT-V</b>  | <b>TRANSITION TESTING</b>       | <b>Classes: 10</b> |
| State, state graphs and transition testing: State graphs, good and bad state graphs, state testing, testability tips.  |                                 |                    |
| <b>Text Book:</b>  |                                 |                    |
| Boris Beizer, -Software Testing Techniques, Dreamtech Press, 2 <sup>nd</sup> Edition, 2003.  |                                 |                    |
| <b>Reference Books:</b>  |                                 |                    |
| <ol style="list-style-type: none"> <li>1. P. C. Jorgenson, -Software Testing: A Craftmen's Approach, Auerbach Publications, 3<sup>rd</sup> Edition, 2013.</li> <li>2. Perry, -Effective Methods of Software Testing, John Wiley, 2<sup>nd</sup> Edition, 1999.</li> <li>3. P. Nageswara Rao, -Software Testing Concepts and Tools, DreamTech Press, 2<sup>nd</sup> Edition, 2007.</li> </ol>   |                                 |                    |
| <b>Web References:</b>   |                                 |                    |
| <ol style="list-style-type: none"> <li>1. <a href="http://www.qatutorial.com/?q=Software_Test_Metrics">http://www.qatutorial.com/?q=Software_Test_Metrics</a></li> <li>2. <a href="http://softwaretestingfundamentals.com/unit-testing/">http://softwaretestingfundamentals.com/unit-testing/</a></li> <li>3. <a href="http://qainsights.com/challenges-in-test-automation/">http://qainsights.com/challenges-in-test-automation/</a></li> <li>4. <a href="http://www.softwaretestinghelp.com/manual-and-automation-testing-challenges/">http://www.softwaretestinghelp.com/manual-and-automation-testing-challenges/</a></li> </ol>   |                                 |                    |
| <b>E-Text Books:</b>   |                                 |                    |
| <ol style="list-style-type: none"> <li>1. <a href="http://www.softwaretestinghelp.com/practical-software-testing-new-free-ebook-download/">http://www.softwaretestinghelp.com/practical-software-testing-new-free-ebook-download/</a></li> <li>2. <a href="http://www.guru99.com/software-testing.html">http://www.guru99.com/software-testing.html</a></li> <li>3. <a href="http://www.fromdev.com/2012/04/8-best-software-testing-books-every-qa.html">http://www.fromdev.com/2012/04/8-best-software-testing-books-every-qa.html</a></li> <li>4. <a href="https://onlinecourses.nptel.ac.in/noc16_cs16/preview">https://onlinecourses.nptel.ac.in/noc16_cs16/preview</a></li> </ol> |                                 |                    |