

NIRMA UNIVERSITY

Institute:	Institute of Technology
Name of Programme:	Master of Computer Application (2-Years Programme)
Course Code:	3MCAD309
Course Title:	Software Testing and Validation
Course Type:	Departmental Elective
Year of Introduction:	2021-22

Credit Scheme

L	T	Practical Component				C
		LPW	PW	W	S	
3	0	0	-	-	-	3

Course Learning Outcomes (CLO):

At the end of the course, students will be able to –

1. define complete software testing life cycle
2. demonstrate understanding of test management process
3. identify the need of automation testing
4. interpret the basic principles of software verification and validation

Syllabus:

Total Teaching hours: 45

Unit	Syllabus	Teaching hours
Unit-I	Introduction to Testing: Testing Introduction, Role of Tester, Testing and Quality, Overview of STLC, Software Testing Life Cycle - V model, SDLC vs STLC, different stages in STLC, document templates generated in different phases of STLC, different levels of testing, different types of testing	04
Unit-II	Static Test Techniques: Static testing reviews, walkthroughs, Basics of test design techniques, Various test categories, test design techniques for different categories of tests. Designing test cases	07
Unit-III	Test Management: Documenting test plan and test case, effort estimation, configuration management, project progress management, test management case study	06
Unit-IV	Defect Management: Test Execution, logging defects, defect lifecycle, fixing / closing defects, tracing defects	06
Unit-V	Test Data Management: Test Data Management –Overview, Needs of Test Data Management, Test Data Types, and Need for Test Data Setup, Test Data Setup Stages, and Test data management Challenges	04
Unit-VI	Basics of Automation testing: Introduction to automation testing, why automation, what to automate, tools available for automation testing	04
Unit-VII	Automation Testing Tool: Basics of Automation testing using Selenium, Introduction to Selenium, using Selenium IDE for automation testing	06

Unit-VIII **Verification and validation of Software:** Software Inspections and Audit, Automated Analysis, Critical systems validation, Five Views of Software Quality, McCall's Quality Factors and Criteria, Quality Factors Quality Criteria, Relationship between Quality Factors and Criteria, Quality Metrics, Various Quality models

08

Self-Study: The self-study contents will be declared at the commencement of semester. Around 10% of the questions will be asked from self-study contents.

Suggested Readings/
References:

1. Rex Black, Managing the Testing Process, John Wiley & Sons
2. Foundations of software testing – by Dorothy Graham, Erik van Veenendaal, Isabel Evans, Rex Black
3. Elfriede Dustin, Implementing Automated Software Testing: How to Save Time and Lower Costs While Raising Quality
4. Ian Sommerville, Software Engineering, Addison – Wesley
5. Roger Pressman, Software Engineering A Practitioner's Approach, McGraw Hill Publication
6. Rajib Mall, Fundamentals of Software Engineering, Prentice Hall of India
7. Ivar Jacobson, Object Oriented Software Engineering A use case Approach, Pearson
8. Selenium.org - <http://docs.seleniumhq.org/docs/>
9. Sagar Naik, Piyu Tripathy, Software Testing and Quality Assurance: Theory and Practice, Wiley.
10. Paul C. Jorgensen, Software Testing - A Craftsman's Approach, CRC Press.
11. Srinivasan Desikan and Gopalaswamy Ramesh, Software Testing, Pearson Education.
12. Louis Tamres, Introducing to Software Testing, Addison Wesley Publications.
13. Glenford J. Myers, The Art of Software Testing, John Wiley & Sons.
14. Robert V. Binder, Testing Object-Oriented Systems: Models Patterns and Tools, Addison Wesley.
15. Daniel Galin, Software Quality Assurance, Pearson Education.

Suggested List of Experiments: -NA-

Suggested Case List: -NA-