NIRMA UNIVERSITY

Institute:	Institute of Technology, School of Technology
Name of Programme:	MTech CSE (Cyber Security)
Course Code:	6CS464ME25
Course Title:	Secured Application Testing and Quality Assurance
Course Type:	Department Elective-III
Year of Introduction:	2025-26

L	T	Practical Component				C
		LPW	PW	W	S	
3	0	2	-	-	-	3

Course Learning Outcomes (CLO):

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

1. explain various security threats in a system (BL2)
2. develop code-based solutions to address security problems (BL3)
3. analyse potential vulnerabilities within the system (BL4)
4. assess security risks in the system. (BL5)

Unit	Contents	Teaching Hours (Total 45)
Unit-I	**Principles of Security Testing:** Introduction to secured application and its testing requirements, apply contemporary formal mathematical modeling techniques to model and analyze the security of a software system, identify project security risks and select risk management strategies, Principles of security testing - Confidentiality, Integrity, Authentication, Authorization, Availability, Non-repudiation, Major focus areas of the security testing- Network Security, System Software Security, Client-side Application Security and Server-side Application Security	08
Unit-II	**Types of Security Testing:** Vulnerability Scanning: automated software to scan a system to detect the known vulnerability patterns, Security Scanning: identification of network and system weaknesses, reducing the defects or risks, Security scanning - manual and automated, Penetration Testing: simulation of the attack from a malicious hacker, to examine for potential vulnerabilities from a malicious hacker that attempts to hack the system, Risk Assessment: risk assessment, testing security risks, and low, medium and high risk, measures to minimize the risk, Use statistical methods to collect and analyze metrics for assessing and improving the security of a product, process, and project objectives	08
Unit-III	**Security Auditing:** Internal inspection of applications and operating systems for security defects, line by line checking of code-static techniques, Dynamic techniques for security auditing	08
Unit-IV	**Data Protection:** Data security concerns, levels of abstraction, data privacy and compliance, security requirements for software systems, designing secured software solutions, quality assurance and strategies, early vulnerability detection	08

Unit-V	**Ethical Hacking and Posture:** Ethical hacking, malicious hacking, identification of the security flaws in the organization system, security scanning, ethical hacking, and risk assessments to provide an overall security posture of an organization. Testing Tools: Benefits of Automation Testing, Random Testing, Bug Bashes and Beta Testing. Test Planning: Test Planning, Test Cases, Bug life cycle	08
Unit-VI	**Software Quality Assurance:** Definition of Quality, Testing and Quality Assurance at Workplace, Test Management and Organizational Structure, Software Quality Assurance Metrics, Quality Management in IT.	05

Self-Study:

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

Suggested Readings/ References:

1. Gary McGraw, Software Security: Building Security, Addison- Wesley
2. Julia H. Allen, Sean Barnum, Robert J. Ellison, Gary McGraw, and Nancy Mead, Software Security Engineering: A Guide for Project Managers, Addison-Wesley
3. Takanen, Ari, Jared D. Demott, Charles Miller, and Atte Kettunen. Fuzzing for software security testing and quality assurance, Artech House
4. Shirasagar Naik, Priyadarshi Tripathy, Software Testing and Quality Assurance: Theory and Practice, Wiley.

Suggested List of Experiments:

Sr. No.	Name of Experiments/Exercises	Hours
1	Develop a program that accepts a program as input and performs a check on coding ethics. Take any programming language of your choice	06
2	Demonstration of various source code analysis tools	06
3	Demonstration of various IDEs to perform the software testing using the preferences option	06
4	Use open-source tools to perform risk assessment like Open-Source Risk Engine, SimpleRisk, Eramba, RA Risk Coverage, PTA Professional, OpenVAS	06
5	Demonstration of Project Risk Manager – Cloud-based version.	06