NIRMA UNIVERSITY

Institute:	Institute of Technology, School of Technology
Name of Programme:	BTech CSE
Course Code:	4CS506ME25
Course Title:	Agile Software Development
Course Type:	Department Elective-IV
Year of Introduction:	2025-26

L	T	Practical Component				C
		LPW	PW	W	S	
3	0	2	-	-	-	4

Course Learning Outcomes (CLO):

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

1. implement continuous integration using essential design principles, refactoring, and version control (BL3)
2. examine customer needs and market conditions, ensuring that the software they develop remains relevant and valuable (BL4)
3. evaluate the significance of integrating agile methodologies and development practices within the business (BL5)
4. develop testing activities seamlessly within the framework of an agile project (BL6)

Unit	Contents	Teaching Hours (Total 45)
Unit-I	**Fundamentals of Agile:** The Genesis of Agile, Introduction and Background, Agile vs. Traditional Waterfall Model, Agile Manifesto and Principles, Overview of Scrum, Extreme programming, Feature-driven development, Agile project management, Design and development practices in Agile projects	09
Unit-II	**Agile Frameworks:** Introduction to Scrum, Project phases, Agile Estimation, Planning game, Product backlog, Sprint backlog, Iteration planning, User story definition, Characteristics and content of user stories, Acceptance tests and Verifying stories, Project velocity, burn down chart, Sprint planning and retrospective, Daily scrum, Scrum roles – Product Owner, Scrum Master, Scrum Team, Introduction of Kanban and compare it with Scrum. Extreme Programming (XP), Lean Software Development	10
Unit-III	**Agile Testing:** The Agile lifecycle and its impact on testing, Test-Driven Development (TDD), Testing user stories - acceptance tests and scenarios, Planning and managing the testing cycle, and Agile risk management.	09
Unit-IV	**Agile Software Design and Development:** Agile design practices, Role of design Principles including Single Responsibility Principle, Open Closed Principle, Liskov Substitution Principle, Interface Segregation Principles, Dependency Inversion Principle in Agile Design, Need and significance of Refactoring.	09
Unit-V	**DevOps in Agile:** Agile Metrics and Performance Measurement, CI/CD (Continuous Integration/ Continuous Development) Scaling	08

Agile for Large Projects (SAFe, LeSS, etc. Agile in Distributed Teams, Agile in Industry Verticals (e.g., Agile in Healthcare). Agile Project Simulation, Research Opportunities in Agile.

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. Ken Schawber, Mike Beedle, *Agile Software Development with Scrum*, Pearson
2. Lisa Crispin, Janet Gregory, *Agile Testing: A Practical Guide for Testers and Agile Teams*, Addison Wesley
3. Robert C. Martin, *Agile Software Development, Principles, Patterns and Practices*, Prentice Hall
4. Alistair Cockburn, *Agile Software Development: The Cooperative Game*, Addison Wesley
5. Mike Cohn, *User Stories Applied: For Agile Software*, Addison Wesley

Suggested List of Experiments:

Sr. No.	Name of Experiments/Exercises	Hours
1	A concise report has to be prepared on comprehending the context and motivating factors for adopting an Agile Approach to Software Development, specifically highlighting the industry's perspective on the necessity of ASD.	02
2	Finalizing a Project System Title, which will be developed by applying the SCRUM Framework in the next lab sessions. List the features and functionalities for the system to be developed. Also, the tools and technologies to be used must be identified. Note: A group of a minimum of 4 and a maximum of 5 students have to be formed to apply Agile principles for understanding Agile-based product development.	04
3	Investigation and Comparative Analysis of SCRUM Project Management Tools, with practical experience using the tool chosen for the development of the system selected by the group.	02
4	Crafting User Stories for the entire project and utilizing the SCRUM Project Management tool to create and securely store them within the tool's repository.	02
5	Evaluating user stories through acceptance tests and scenarios, establishing clear Acceptance Criteria for each User Story developed in Practical-4. Additionally, the complexity of these user stories can be assessed by leveraging an online Planning Poker tool.	02

6	Creating a Product Backlog and planning Sprints for the chosen project's definition. Ensuring product verification and validation for every Sprint result. Carrying out key SCRUM events such as Sprint Execution, Sprint Review, and Sprint Retrospective. Please note that we'll be organizing 1-week sprints and conducting corresponding Sprint Reviews and Retrospectives accordingly.	04
7	A concise report is to be crafted, highlighting the distinctions in testing between Agile-based Software Development and conventional approaches. This report will also encompass a comparative analysis of different testing tools associated with SCRUM, alongside practical experience in Agile-based testing methodologies.	04
8	Formulating test cases and diligently executing them for the chosen project's definition and the associated implementation work.	04
9	Engage in Behavioral Driven Development (BDD), which involves conducting comprehensive feature-level testing for the ongoing project.	02
10	Conduct the product deployment and perform thorough acceptance testing. Simultaneously, create a comprehensive project user manual encompassing detailed documentation for all modules, along with accompanying screenshots and descriptive explanations.	04