

**NIRMA UNIVERSITY**  
**Institute of Technology**  
**B. Tech. Computer Science and Engineering**  
**Semester – VI**  
**Department Elective-I**

L	T	P	C
2	0	2	3

<b>Course Code</b>	2CSDE55
<b>Course Title</b>	Agile Software Development

**Course Outcomes:**

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

1. appraise the business value of adopting Agile approaches and development practices
2. apply design principles, refactoring version control and continuous integration to achieve Agility
3. implement testing activities within an Agile project using various testing strategies.

**Syllabus**

**Teaching  
Hours: 30  
06**

**Unit I**

**Fundamentals of Agile:** The Genesis of Agile, Introduction and background, Agile Manifesto and Principles, Overview of Scrum, Extreme programming, Feature driven development, Lean Software Development, Agile project management, Design and development practices in Agile projects

**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, Scrum case study, Tools for Agile project management, Introduction of Kanban and compare it with Scrum

**Unit III**

**Agile Testing:** The Agile lifecycle and its impact on testing, Test-Driven Development (TDD), xUnit framework and tools for TDD, Testing user stories - acceptance tests and scenarios, Planning and managing testing cycle, Exploratory testing, Risk based testing, Regression tests, Test Automation, Tools to support the Agile tester.

**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, Refactoring Techniques, Continuous Integration, Automated build tools, Version control.

12

05

07



**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.

**Laboratory Work:**

Laboratory work will be based on the above syllabus with minimum 5 experiments to be incorporated.

**Suggested Readings<sup>^</sup>:**

1. Ken Schwaber, 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

L=Lecture, T=Tutorial, P=Practical, C=Credit

---

<sup>^</sup>this is not an exhaustive list