

## **BCSE0053: AGILE SOFTWARE DEVELOPMENT**

**Objective:** The course objective is to teach the fundamental principles and practices associated with each of the Agile development methods. A variety of Agile methods will be described, but the focus will be on Scrum and Extreme Programming.

**Prerequisite:** Exposure of Software Engineering Principles.

**Credits: 03**

**L-T-P-J: 3-0-0-0**

<b>Module No.</b>	<b>Content</b>	<b>Teaching Hours</b>
I	<p><b>Introduction:</b> The Genesis of Agile, Introduction and background, Agile Manifesto and Principles, Overview of Scrum, Extreme Programming, Feature Driven development, Feature Driven development, Agile project management, Design and development practices in Agile projects, Test Driven Development, Continuous Integration, Refactoring, Pair Programming, Simple Design, User Stories, Agile Testing, Agile Tools.</p> <p><b>Agile Scrum Framework:</b> 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 charts, Sprint planning and retrospective, Daily scrum, Scrum roles – Product owner, Scrum Master, Scrum Team, Scrum case study, Tools for Agile project management.</p> <p><b>Agile Software Design:</b> Agile design practices, Role of design Principles including Single Responsibility Principles, Open Closed Principles, Liskov Substitution Principles, Interface Segregation Principles, Dependency Inversion Principle in Agile Design.</p>	20
II	<p><b>Agile Software Development:</b> Need and significance of Refactoring, Refactoring Techniques, Continuous Integration, Automated build tools, Version control. Current researches in Agile software development.</p> <p><b>Agile Testing:</b> 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.</p> <p><b>Industry Trends:</b> Market scenario and adoption of Agile, Agile ALM, Roles in an Agile project, Agile applicability, Agile in Distributed teams, Business benefits, Challenges in Agile, Risks and Mitigation, Agile projects on Cloud, Balancing Agility with Discipline, Agile rapid development technologies.</p>	20

**Text Books:**

- Robert C. Martin, Agile Software Development: Principles, Patterns & Practices, Prentice Hall, 2002.

**References Books:**

- Ken Schwaber & Mike Beedle, Agile software Development with Scrum, Pearson, 2008.
- Lisa Crispin & Janet Gregory, Agile Testing: A Practical Guide for Testers and Agile Teams, Addison Wesley, 2008.
- Alistair Cockburn, Agile Software Development: The Cooperative Game, Addison Wesley, 2006.

**Outcome:** After the completion of the course, the student will:

- CO1- Understand the significance of Agile Methodologies in software development.
- CO2- Compare and contrast the different Agile methods.
- CO3- Determine the suitability of Agile methods for a particular Project.
- CO4-Evaluate how well a project is following Agile principles, and assist the project to become more Agile (where appropriate).
- CO5- Understand the relationship between the customer and the development team in Agile projects and the responsibilities of both communities.