

NIRMA UNIVERSITY

Institute:	Institute of Technology
Name of Programme:	Master of Computer Application (2-Years Programme)
Course Code:	3MCAD362
Course Title:	Agile Software Development
Course Type:	Departmental Elective
Year of Introduction:	2021-22

Credit Scheme

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

Course Learning Outcomes (CLO):

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

1. summarize the Agile design, development practices and recent trends in Industry
2. apply design principles and refactoring to achieve Agility
3. analyse Agile project management practices
4. test the application for unit tests using Test Driven Development

Syllabus:

Total Teaching hours: 45

Unit	Syllabus	Teaching hours
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, Test Driven Development, Continuous Integration, Refactoring, Pair Programming, Simple Design, User Stories, Agile Testing, Agile Tools	05
Unit-II	Agile Scrum Framework: 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	12
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	12
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

Unit-V **Industry Trends:** 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 04

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. 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

Suggested List of Experiments:	Sr. No.	Title	Hours
	1	To perform testing select one project definition and write down the test cases (at least 5) in excel sheet for the same.	02
	2	To perform the unit test, develop unit test code and examine it on the test cases which you have been written in practical number 1.	02
	3	To install and configure the Jenkins software with all plugins like git bash, continuous deployment tools.	02
	4	To perform unit test, implement the java code with unit test case and illustration version control mechanics using Jenkin's plugin like git or subversion.	02
	5	To demonstrate how continuous development and integration channels works, design the code for the student registration system with student id and password as information and deploy on Jenkin.	04
	6	To develop the code to generate the matrix which is showing code analysis.	02
	7	To show how effective and bug free code you have by using matrix evaluation plugin, study the different agile software tools like AgileFant, OrangeScrum and IceScrum and compare them.	04
	8	To write a user stories for restaurant management system with consideration of 3 users are going to use the system. Write down their complexity using the PlanningPoker online tool.	04

- | | | |
|----|---|----|
| 9 | To implement agile framework on restaurant management system. Show all the artifacts of agile. Use any open source tool for the same. | 04 |
| 10 | To develop the scenario based user story on canteen management system. Use any open source tool to demonstrate it. | 04 |

Suggested Case List: -NA-