

AGILE SOFTWARE DEVELOPMENT

(Professional Elective –IV)

Course Code	20IT4702C	Year	IV	Semester	I
Course Category	PE - IV	Branch	IT	Course Type	Theory
Credits	3	L-T-P	3-0-0	Prerequisites	Software Engineering
Continuous Internal Evaluation :	30	Semester End Evaluation:	70	Total Marks:	100

Course Outcomes		Blooms Taxonomy Level
Upon Successful completion of course, the student will be able to		
CO1	Understand the basics of Agile methods in various development environments.	L2
CO2	Apply different software development processes in real situations	L3
CO3	Use Agile tools for software development processes in different scenarios	L3
CO4	Analyze different software development methods	L4

Contribution of Course Outcomes towards achievement of Program Outcomes & Strength of Correlations (H:High, M:Medium, L:Low)														
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3												3	
CO2	3										3			3
CO3			3		3						3			
CO4	3		3								3			3

Syllabus		
Unit No	Contents	Mapped CO
I	INTRODUCTION: What is Agile? The Agile manifesto, agile methods, XP: Extreme Programming, DSDM, SCRUM, Feature-Driven Development, modeling misconceptions, agile modeling, tools of misconceptions, updating agile models.	CO1
II	Extreme Programming: Introduction, core XP values, the twelve XP practices, about extreme programming? Planning XP projects, test first coding, making pair programming work.	CO1 CO2 CO4
III	Agile Modelling and XP: Introduction, the fit, common practices, modelling specific practices, XP objections to agile modelling, agile modelling and planning XP projects, XP implementation phase.	CO1 CO2 CO4
IV	Feature-Driven Development: Introduction, incremental software development, Regaining Control: The motivation behind FDD, planning an iterative project, architecture centric, FDD and XP.	CO1 CO2 CO4

V	Agile Methods with RUP and PRINCE2 and Tools and Obstacles: Agile modeling and RUP, FDD and RUP, agile methods and prince2, tools to help with agile development, Eclipse: An agile IDE, obstacles to agile software development, management intransigence, the failed project syndrome, contractual difficulties, familiarity with agility.	CO1 CO3 CO4
----------	---	--

Learning Resources
Text Books
1. Craig Larman, Agile and Iterative Development, Addison-Wesley, Pearson Education, 2004.
References
1. Agile Software Development, Principles, Patterns and Practices, Pearson New International Edition, 2013. 2. Pearson, Robert C. Martin, Juli, James Shore, Chromatic, the Art of Agile Development, O'Reilly Media, 2013.
E-Resources and other Digital Material
1. https://www.udacity.com/course/agile-software-development-nanodegree--nd144