



# INSTITUTE OF AERONAUTICAL ENGINEERING

(Autonomous)

Dundigal - 500 043, Hyderabad, Telangana

## COURSE CONTENT

AGILE METHODOLOGIES							
<b>I Semester: CSE</b>							
Course Code	Category	Hours / Week			Credits	Maximum Marks	
BCSE07	<b>Elective</b>	<b>L</b>	<b>T</b>	<b>P</b>	<b>C</b>	<b>CIA</b>	<b>SEE</b>
		3	0	0	3	40	60
<b>Contact Classes: 45</b>	<b>Total Tutorials: Nil</b>	<b>Total Practical Classes: Nil</b>			<b>Total Classes: 45</b>		

### I. COURSE OVERVIEW:

This course provides a structured, in-depth exploration of modern agile software development, with a special focus on the methodology Extreme Programming (XP). Beginning with the “why” of agility and organizational success, it progresses through collaboration, release practices, planning, and developing high-quality software via incremental, iterative methods. Students will learn how to adopt and practice XP in real-world scenarios, covering team dynamics, technical practices (like TDD, refactoring, continuous integration), and planning & release strategies. The objective is to equip learners with the mindset, methods and tools to deliver reliable software rapidly and adaptively in dynamic project environments.

### II. COURSE OBJECTIVES:

#### The students will try to learn:

- I. The principles, values, and need for Agile development, emphasizing organizational success, adaptability, and customer satisfaction beyond meeting deadlines.
- II. The key practices of Extreme Programming (XP) — including pair programming, test-driven development, refactoring, and continuous integration — to produce high-quality, incremental software.
- III. Collaborative and communication skills essential for Agile teamwork, such as real customer involvement, iteration planning, and effective retrospectives.
- IV. How to plan, execute, and evaluate Agile projects through iteration planning, risk management, release planning, and continuous improvement of processes and team agility.

### III. COURSE OUTCOMES:

#### After successful completion of the course, students should be able to:

- CO1 Explain the fundamental principles, values, and practices of Agile development and Extreme Programming (XP).
- CO2 Differentiate between traditional software development approaches and Agile methodologies in terms of adaptability, collaboration, and delivery.
- CO3 Apply key XP practices such as pair programming, test-driven development, refactoring, and continuous integration to develop high-quality software.
- CO4 Collaborate effectively in Agile teams by involving real customers, maintaining coding standards, and conducting iteration reviews and retrospectives.
- CO5 Plan and manage Agile projects through effective release planning, iteration planning, risk management, and story estimation.
- CO6 Evaluate and improve team agility and development processes through feedback, root cause analysis, and continuous process refinement.

#### **IV. COURSE CONTENT:**

##### **MODULE-I: INTRODUCTION EXTREME PROGRAMMING (XP) - AGILE DEVELOPMENT (9)**

Why Agile - Understanding Success, Beyond Deadlines, Importance of Organizational Success, Introduction to Agility How to Be Agile - Agile methods, Don't make your own method, Road to mastery Understanding XP (Extreme Programming) - XP life cycle, XP team, XP Concepts Adopting XP - Knowing whether XP is suitable, Implementing XP, assessing Agility Practicing XP - Thinking - Pair Programming, Energized work, Informative Workspace, Root cause Analysis, Retrospectives

##### **MODULE-II: COLLABORATING (9)**

Trust, Sit together, Real customer involvement, Ubiquitous language, meetings, coding standards, Iteration demo, Reporting.

##### **MODULE-III: RELEASING (9)**

Bugfree Release, Version Control, fast build, continuous integration.

Collective ownership, Documentation

##### **MODULE-IV: PLANNING (9)**

Version, Release Plan, Risk Management, Iteration Planning, Slack, Stories, Estimating.

##### **MODULE-IV: DEVELOPING (9)**

Incremental requirements, Customer tests, Test driven development, Refactoring, Incremental design and architecture, spike solutions, Performance optimization, Exploratory testing

#### **V TEXT BOOKS:**

1. The art of Agile Development, James Shore and Shane Warden, 11th Indian Reprint, O'Reilly, 2018.

#### **VI. REFERENCE BOOKS:**

1. Learning Agile, Andrew Stellman and Jennifer Greene, O'Reilly, 4th Indian Reprint, 2018.
2. Practices of an Agile Developer, Venkat Subramaniam and Andy Hunt, SPD, 5th Indian Reprint, 2015.
3. Agile Project Management - Jim Highsmith, Pearson Low price Edition 2004.

#### **VII. Web References:**

1. <https://www.agilealliance.org/glossary/continuous-integration/>
2. <https://www.atlassian.com/agile/release-management>
- 3 <https://martinfowler.com/articles/continuousIntegration.html>

#### **VIII. E-BOOKS:**

1. eBook - Extreme Programming Explained by Kent Beck · OverDrive: Free ebooks, audiobooks & movies from your library.
2. [https://www.overdrive.com/media/613819/agile-estimating-and-planning?utm\\_source=chatgpt.com](https://www.overdrive.com/media/613819/agile-estimating-and-planning?utm_source=chatgpt.com)

#### **IX. MATERIALS ONLINE**

1. Course template
2. Tutorial question bank
3. Tech talk topics
4. Open, Ended experiments
5. Definitions and terminology
6. Assignments

7. Model question paper, I
8. Model question paper, II
9. Lecture notes
10. Power Point presentation
11. E-Learning Readiness Videos (ELRV)