

Course Code- **MC1105**Course Title- **Software Engineering**

Teaching Scheme				Examination Scheme			
Lectures	Tutorials	Practicals	Credits	CT	TA	ESE	Total
04	-	-	04	20	20	60	100

Course Objectives

- 1 To understand fundamental concepts of software engineering principles.
- 2 To understand and transform the basic customer requirements in to a feasible technical solution.
- 3 To understand how to design and implement software with the help of various standard prototypes and models.

Course Outcomes- After studying this course, students will be able to:

- CO 1 Gather, Categorise and Interpret the user requirements and prepare SRS (System Requirement Specification) document.
- CO 2 Transform SRS into design and then into ready to execute software to satisfy complete user needs accurately.
- CO 3 Be agile software developers with a comprehensive set of skills appropriate to the needs of the dynamic global computing-based society.
- CO 4 Capable of team and organizational leadership in computing project settings, and have a broad understanding of ethical application of computing-based solutions to societal and organizational problems.
- CO 5 Acquire skills and knowledge to provide software maintenance to customer.

Course Contents

Unit No	Detailed Contents	Contact Hours
1	Introduction to Software Engineering Basic Terminologies: What is Software & its Characteristics, requirements, specifications, design, testing, validation, evolution and project management, Software Crises and Myths, Software Engineering as a Layered Technology, Software Development Lifecycle, Agile view of Process	8
2	Software Process, requirements and specification: Software Process, Process Framework, Generic and Umbrella activities, Process Patterns, Process Assessment, Various Process Models, Requirement Engineering -inception, elicitation, elaboration, negotiation, specification, validation & management, Analysis Modelling Techniques - UML Diagrams	8
3	Software Design: Translating analysis model into design model, Fundamental design Concepts and Principles, , Design Process, Quality Attributes, Software Architecture & Architectural Styles.	8
4	Software Testing, Validation & Maintenance: Testing Strategy for Conventional Software, Testing Fundamentals, Test	8

Plan Creation and Test Cases Generation, Unit and Integration Testing, Black-box and White-box Testing Techniques, Validation, and System Testing.

5 **Software Evolution:**

Software Maintenance- Corrective, Adoptive, Perfective, Structured and Un-Structured, The Associated Disciplines and the Role and the Nature of the Configuration Management, Characteristics of Maintainable Software, Software Re-use strengths and weaknesses, Re-engineering. 8

Text Books

- 1 Software Engineering” – Roger S. Pressman, TMH
- 2 K.K.Aggarwal , Yogesh Singh , “Software Engineering” , New Age International Publishers
- 3 Shrinivasan, Gopalaswamy, “Software Testing” – Pearson Education

Reference Books

- 1 William Perry, “Effective Methods for Software Testing”, John Wiley & Sons, New York, 1995.
- 2 CemKaner, Jack Falk, Nguyen Quoc, “Testing Computer Software”, Second Edition, Van Nostrand Reinhold, New York, 1993.
- 3 BorisBeizer, “Software Testing Techniques”, Second Volume, Second Edition, Van Nostrand Reinhold, New York, 1990.
- 4 LouiseTamres, “Software Testing”, Pearson Education Asia, 2002

E Books/ Online learning material

- 1 NPTEL - <https://nptel.ac.in/courses/106/105/106105087/>
- 2 SWAYAM- https://swayam.gov.in/nd1_noc19_cs69/preview
- 3 COURSERA-<https://www.coursera.org/courses?query=software%20engineering>

Assessment Table:

Assessment Tool	Course Outcomes				
	CO1	CO2	CO3	CO4	CO5
Evaluation I (Class Test) 20 Marks	10	10	-	-	-
Teachers Assessment 20 Marks	5	5	-	5	5
ESE Assessment 60 Marks	18	12	12	6	12

Teaching Strategies:

1. Use of Conventional Black board/White board
2. Use of Power Point Presentations
3. Group Discussion
4. Group and Individual activities/tasks
5. Class room question-answer sessions