

25UCST412	MANAGEMENT OF SOFTWARE SYSTEMS	Category	L	T	P	R	Credit	Year of Introduction
		PEC	3	0	0	0	3	2025

Preamble: Software Engineering is a core discipline in Computer Science that deals with the systematic development, operation, and maintenance of reliable software systems. This course covers software development life cycle models, requirement engineering, system design, testing, DevOps, and project management. It includes both theoretical concepts and practical approaches such as UML modelling, estimation, risk management, Agile and Micro services. The course also highlights quality assurance, professional ethics, and teamwork.

Prerequisite: Nil

Course Outcomes: After the completion of the course, the student will be able to:

CO	Course Outcomes	Knowledge Level
CO1	Explain software engineering principles, process models and requirement engineering concepts.	K2 Understanding
CO2	Apply architectural patterns, design principles, UML modeling techniques, and design patterns to develop structured software systems.	K3 Apply
CO3	Apply coding standards, testing strategies, DevOps practices and maintenance techniques for ensuring software quality.	K3 Apply
CO4	Demonstrate estimation models for project management with risk analysis and quality standards.	K3 Apply

Mapping of course outcomes with program outcomes and program specific outcomes

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	–	–	–	–	1	–	–	–	1	2	1
CO2	3	3	3	1	1	–	–	–	–	–	1	2	3
CO3	3	2	2	1	3	–	–	–	–	–	1	2	3
CO4	3	2	1	–	3	–	–	–	–	3	1	1	2
Avg	3	2.25	2	1	2.33	–	1	–	–	3	1	1.75	2.25

Mapping of course outcomes with Knowledge and Attitude Profile (WK)

	WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
CO1	*	*	*	*				*	*
CO2	*	*	*	*	*	*		*	
CO3	*	*	*	*	*	*		*	
CO4	*	*	*	*	*	*		*	

Wks Mapped with the course

WK1	WK2	WK3	WK4	WK5	WK6	WK7	WK8	WK9
*	*	*	*	*	*		*	*

Mark Distribution:

Total Marks	CIE Marks	ESE Marks	ESE Duration
100	40	60	2.5 hrs

Continuous Internal Evaluation Pattern:

Attendance	: 5 marks
Continuous Assessment Test	: 20 marks
Continuous Assessment Assignment	: 15 marks

SYLLABUS**Module 1 (11 hrs)**

Introduction to Software Engineering, Ethics and Professional Practices, Software process models - Waterfall, Incremental Models, Spiral Models, Agile Model, Agile Manifesto, Scrum Overview, Process activities – Specification, Design and Validation, Requirement Types (Functional & Non functional), Requirement Elicitation, Requirement Validation & Change Management, Feasibility Analysis & Traceability Matrix, SRS Document & Structure, Case study (Library Management System).

Module 2 (11 hrs)

Design Concepts & Functional Independence, Coupling & Cohesion, Software Architecture & Importance, Architectural Patterns (Layered, Client-Server, Repository), Component & Connector, Publish-Subscribe, UML – Use Case Diagram, UML – Class Diagram, UML – Sequence Diagram, UML – State Diagram, Design Patterns – Creational, Structural & Behavioral Patterns + Ariane Case Study.



Module 3 (10 hrs)

Coding Guidelines & Review Techniques, Debugging Methods & Cost of Defects, Unit Testing & Integration Testing, System Testing & Validation Testing, White Box Testing, Black Box & Regression Testing, Test Automation & TDD, Open Source & Licensing, DevOps – CI/CD & Code Management, Software Maintenance & Boehm Models.

Module 4 (10hrs)

Project Metrics – LOC, FP, Object Points, COCOMO Cost Estimation, Risk Management, Project Planning & Scheduling (Gantt Chart), Agile Project Management – Scrum, Configuration & Change Management, Software Quality & SQA Tasks, ISO, CMMI & Six Sigma, Micro Services Architecture & Deployment.

Completion of an industrial visit or industrial training is desirable for this course

Text Books:

1. Sommerville, I. (2025). *Software Engineering* (10th ed.). Pearson Education.
2. Pressman, R. S., & Maxim, B. (2023). *Software Engineering: A practitioner's approach* (9th ed.). McGraw-Hill Education.
3. Laplante, P. A., & Kassab, M. (2022). *Requirements Engineering for Software and Systems* (4th ed.). CRC Press.

Reference Books:

4. Chandramouli, S., & Dutt, S. (2015). *Software project management*. Pearson Education.
5. Sommerville, I. (2020). *Engineering software products: An introduction to modern software engineering* (1st ed.). Pearson Education.
6. Brooks, F. P., Jr. (1995). *The mythical man-month: Essays on software engineering* (Anniversary ed.). Addison-Wesley.

END SEMESTER EXAMINATION -MODEL QUESTION PAPER

Course Code	25UCST412
Course Name	MANAGEMENT OF SOFTWARE SYSTEMS
Max. Marks: 60	Duration: 2.5 Hours

Part A

(Answer all questions. Each question carries 3 marks)

Qn No.	Question	Marks	CO	BL
1	Explain the phases of the Software Development Life Cycle (SDLC).	3	1	L2
2	Differentiate between functional and non-functional	3	1	L2

	requirements.			
3	A college management system requires separate modules for Student Management, Examination, and Fee Processing. Apply the concept of Layered Architecture and explain how these modules can be organized into layers.	3	2	L3
4	An online shopping application allows customers to place orders and receive notifications about order status changes. Draw a Sequence Diagram showing the interaction between Customer, Shopping System, and Notification Service.	3	2	L3
5	A developer modifies a payment module in an e-commerce application. Apply an appropriate testing technique to ensure that the modification has not affected existing functionalities. Justify your choice.	3	3	L3
6	A software team follows Test-Driven Development (TDD) while developing a login module. Illustrate the sequence of activities performed in TDD for implementing the module.	3	3	L3
7	A software project is estimated to have 20,000 LOC. Apply the COCOMO Basic Model (Organic Mode) to estimate the effort required for development.	3	4	L3
8	A development team is managing a project using Scrum. Apply Scrum practices and explain how the team would handle a change request received during a Sprint.	3	4	L3
Part B				
<i>(Answer any one full question from each Module. Each carry 9 marks)</i>				
Module I				
Qn No.	Question	Marks	CO	BL
9.a	Explain the Waterfall, Spiral, Incremental and Agile process models. Compare their advantages and limitations.	5	1	L2
9.b	Describe requirement elicitation, requirement validation and change management activities with suitable examples.	4	1	L2
OR				
10.a	Explain the structure of an SRS document and discuss the importance of traceability matrix.	5	1	L2
10.b	Discuss Scrum roles, artifacts and ceremonies used in Agile development.	4	1	L2
Module II				
Qn No.	Question	Marks	CO	BL
11.a	Design a Use Case Diagram and Class Diagram for a Library Management System. Clearly identify actors, classes and relationships.	5	2	L3
11.b	Describe various software architectural patterns such as Client-Server, Repository architectures with suitable examples.	4	2	L2
OR				

12.a	Design a Layered Architecture for an Online Banking System and explain the role of each layer.	5	2	L3
12.b	Explain the Publish-Subscribe architectural pattern and its applications	4	2	L2
Module III				
Qn No.	Question	Marks	CO	BL
13.a	Apply white-box testing techniques to determine test paths and test cases for a given program.	5	3	L3
13.b	Describe the Boehm maintenance model and its significance in software evolution.	4	3	L2
OR				
14.a	An organization plans to maintain a legacy software system that requires adaptation to a new operating environment. Apply appropriate software maintenance techniques and justify your choice.	5	3	L3
14.b	Describe the role of DevOps in modern software development and deployment	4	3	L2
Module IV				
Qn No.	Question	Marks	CO	BL
15.a	A software project has identified risks related to schedule delays, cost overruns, and technical uncertainties. Apply the risk management process to analyze and mitigate these risks.	5	4	L3
15.b	Discuss the importance of ISO 9001 in software quality management.	4	4	L2
OR				
16.a	A software organization plans to improve its process quality. Illustrate how CMMI and Software Quality Assurance (SQA) practices can be applied to achieve this objective.	5	4	L3
16.b	Explain the role of CMMI in software process improvement.	4	4	L2