Software Engineering and Project Management			
Course Code	**22CSE51**	CIE Marks	50
Course Type (Theory/Practical/Integrated)	Theory	SEE Marks	50
		Total Marks	100
Teaching Hours/Week (L:T:P)	3:0:0	SEE	3 Hours
Total Hours	40 Hours	Credits	03

Course Learning Objectives: The objective of the course is to

- Outline software engineering principles and activities involved in building large software programs
- Describe the process of requirements gathering, requirements classification, requirements specification and requirements validation
- Analyze, design, implement, verify, validate, implement, apply, and maintain software systems or parts of software systems
- Describe the intricacies involved in software evolution and the need for software testing.

Module-1 Phases and Life cycle models of Software Development (8 hours)

Introduction: Professional software development; Software engineering: ethics and case studies **Software processes**: Software processes Models; The waterfall model, Incremental development, Integration and configuration.
Applications: Identifying phases and Life cycle models of Software Development
Text book 1: Chapter 1: 1.1-1.3 Chapter 2: 2.1

Module-2 Software Process activities, Agile software Development and Requirements Analysis (8 hours)

Process activities: Software specification, Software design and implementation, Software validation, Software evolution.
Agile Software development: Agile methods, Agile development techniques, Extreme Programming, Agile project Management
Software requirements: Functional and non-functional requirements; Requirements engineering Process, Requirement elicitation, Requirement specification, Requirement Validation and management
Applications: Systematically identifying the requirements and using agile method for software development
Text book 1: Chapter 2: 2.2 Chapter 3: 3.1-3.3, Chapter 4: 4.1-4.6

Module-3 System Modeling(8 hours)

System models: Context models; Interaction models; sequence diagram Structured methods; class diagram, generalization, Aggregation. Behavioral Models-Data driven modeling; Event driven modeling, Model driven engineering
Applications: Concept learning to develop system models based on software requirements.
Text book 1: Chapter 5: 5.1-5.5

Module-4 Software Design and Testing (8 hours)

Software Design: Object-oriented design using the UML, Design Patterns, Implementation Issues.
Software testing: Development testing, Test-driven development, Release testing, User testing.
Applications: Selecting the right method for testing software and maintaining the system.
Text book 1: Chapter 7: 7.1-7.3 Chapter 8: 8.1-8.4

Module-5 Evolution of Software and Project Management (8 hours)

Software evolution: Evolution processes, Legacy system management.
Project management: Risk management, Project planning: Software pricing, Plan-driven development, Quality management: Software quality, Software standards.

Applications: Managing a software by analyzing the risk involved and maintaining its quality while undergoing a change.
Text book 1: Chapter 9: 9.1-9.2 Chapter 22: 22.1 Chapter 23: 23.1-23.2 Chapter 24: 24.1-24.2
Additional Resources: Software development tools: Jira, Eclipse IDE, GitHub, Docker

Course Outcomes: At the end of the course the student will be able to:	
22CSE51.1	Describe software process models and apply them in software development scenarios.
22CSE51.2	Describe software process activities and develop a plan for requirement engineering for agile method of software development
22CSE51.3	Apply object orientation and modeling constructs to design modeling diagrams for software systems.
22CSE51.4	Differentiate between various software testing methods and select the right method for testing a software.
22CSE51.5	Apply the principals involved in software evolution while maintaining software and describe the processes involved in project planning and quality management.
22CSE51.6	Function effectively in teams to develop software specification document, system models, and test cases while implementing a systematic approach to problem solving.

Sl. No.	**Title of the Book**	**Name of the Author/s**	**Name of the Publisher**	**Edition and Year**
Textbooks				
1	Software Engineering	Ian Somerville	Pearson Education Ltd	10th Edition, 2017
Reference Books				
1	Software Engineering: A Precise Approach	Pankaj Jalote	Wiley	3rd Edition, 2010
2	Object Oriented Modelling and design with UML	Michael Blaha, James Rumbaugh	Pearson Education	2nd Edition, 2007
3	Software Engineering-A Practitioners approach	Roger S. Pressman	Tata McGraw Hill	9th Edition, 2023

Web links and Video Lectures (e-Resources):

- **Software Engineering Essentials:** https://www.edx.org/course/software-engineering-essentials
- **Basics to Advanced fundamentals of SE:** https://www.udemy.com/the-complete-software-engineering-from-basics-to-advanced/
- **Agile Software Development:** https://www.udemy.com/fundamentals-of-agile-software-development/
- **Scrum Master Tales:** https://www.udemy.com/scrumtalesstoriesfromascrummastersdiary/
- **Test your agile and scrum knowledge:** https://www.udemy.com/test-your-agile-and-scrum-knowledge/?couponCode=DISCUDEMY.COM

Course Articulation Matrix

Course Outcomes (COs)	Program Outcomes (POs)													
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
22CSE51.1	3		1				1							1
22CSE51.2	3			1			1							
22CSE51.3		2	1	2										1
22CSE51.4		2	1	2	1				1	1				1
22CSE51.5			1	2	1									1
22CSE51.6			1					2	2	3	1			

1: Low 2: Medium 3: High