



**B. Tech.  
Semester III**

**SOFTWARE ENGINEERING  
CE4013**

**Effective From July-2022**

**Syllabus version: 1.00**

Subject Code	Subject Title	Teaching Scheme			
		Hours		Credits	
		Theory	Practical	Theory	Practical
CE4013	Software Engineering	3	0	3	0

Subject Code	Subject Title	Theory Examination Marks		Practical Examination Marks	Total Marks
		Internal	External	CIE	
CE4013	Software Engineering	40	60	0	100

#### Objectives of the course:

- To deliver the foundation of large scale software development processes and modeling.
- To unfold different aspects of software design and software testing.

#### Course outcomes:

Upon completion of the course, the student shall be able to,

C01: Understand software engineering and processes

C02: Explore software modeling and requirement engineering

C03: Explore concepts of software design and user architectural design

C04: User component level and user interface design aspects

C05: User pattern based design aspects and apply design aspects of web applications

C06: Understand and apply software quality strategies and testing

Sr. No.	Topics	Hours
<b>Unit – I</b>		
<b>1</b>	<b>Introduction to Software Engineering and Software processes:</b> The nature of software, The Unique nature of WebApps, The software process, Software Engineering practice, Software myths, A generic process model, Process assessment and improvement, prescriptive process models, Specialized process models, The unified process model, Personal and Team process models, Process	9

	Technology, Product and Process. Agile development, The Agility, Agile Process, Extreme programming, Agile process models.	
<b>Unit – II</b>		
<b>2</b>	<b>Software Modeling:</b> Principles of software modeling, Requirement engineering, Eliciting requirements, Developing use cases, Building the requirements model, Negotiating requirements, Validating requirements, Requirements analysis, Scenario-based modeling, UML models, Data modeling concepts, Class based modeling, Requirement modeling strategies, Flow-oriented modeling, Creating a behavioral model, Patterns for requirements modeling, Requirements modeling for WebApps.	<b>9</b>
<b>Unit – III</b>		
<b>3</b>	<b>Software Design Concepts and Architectural Design:</b> The design process, Design concepts, The design model Software architecture, Architectural styles, Architectural design, Assessing alternative architectural designs, Architectural mapping using data flow.	<b>5</b>
<b>Unit – IV</b>		
<b>4</b>	<b>Component Level and User Interface Design:</b> Designing class-based components, Conducting component-level design, Component-level design for WebApps, Designing traditional components, Component-based development. The golden rules of user interface design, User interface analysis and design, Interface analysis, Interface analysis steps, WebApps interface design, Design evaluation.	<b>6</b>
<b>Unit – V</b>		
<b>5</b>	<b>Pattern-Based Design:</b> Design patterns, Pattern-based software design, Architectural patterns, Component-level design patterns, User interface design patterns, WebApp design patterns.  <b>Web Application Design:</b> WebApp design quality, Design goals, A design pyramid for WebApps, WebApp interface design, Aesthetic design, Content design, Architecture design, Navigation design, Component-level design, Object Oriented Hypermedia Design Method (OOHDM).	<b>8</b>
<b>Unit – VI</b>		

6	<b>Software Quality and Testing:</b> Software quality and quality dilemma, Achieving software quality, Introduction to Software review techniques, Introduction to software quality assurance, ISO 9000 quality standards, A strategic approach to software testing, Strategic issues, Test strategies for object oriented softwares, Test strategies for WebApps, Validation testing, System testing, The art of debugging, White box testing, Basic path testing, Control structure testing, Balck-box testing, Model based testing, Object oriented test strategies and methods.	8
---	--	---

**Text Book:**

1. Roger S. Pressman - "Software Engineering – A Practitioner's Approach", 7th Edition, McGrawHill.

**Reference books:**

1. Sommerville - "Software Engineering", Pearson Education.
2. Rajib Mall - "Software Engineering", PHI.
3. Ghezzi, Jazayeri, Mandrioli - "Fundamentals of Software Engineering", Pearson Education.
4. Software Engineering Body of Knowledge (SWEBOK V3.0), IEEE Computer Society.

**Course objectives and Course outcomes mapping:**

- To deliver the foundation of large scale software development processes and modeling: CO1, CO2, and CO3
- To unfold different aspects of software design and software testing: CO4, CO5, and CO6

**Course units and Course outcomes mapping:**

Unit No.	Unit Name	Course Outcomes					
		CO1	CO2	CO3	CO4	CO5	CO6
1	Introduction to Software Engineering and Software processes	✓					
2	Software Modeling		✓				
3	Software Design Concepts and Architectural Design			✓			
4	Component Level and User Interface Design				✓		
5	Pattern-Based Design and Web Application Design					✓	
6	Software Quality and Testing						✓

## **Programme outcomes:**

- PO 1: Engineering knowledge: An ability to apply knowledge of mathematics, science, and engineering.
- PO 2: Problem analysis: An ability to identify, formulates, and solves engineering problems.
- PO 3: Design/development of solutions: An ability to design a system, component, or process to meet desired needs within realistic constraints.
- PO 4: Conduct investigations of complex problems: An ability to use the techniques, skills, and modern engineering tools necessary for solving engineering problems.
- PO 5: Modern tool usage: The broad education and understanding of new engineering techniques necessary to solve engineering problems.
- PO 6: The engineer and society: Achieve professional success with an understanding and appreciation of ethical behavior, social responsibility, and diversity, both as individuals and in team environments.
- PO 7: Environment and sustainability: Articulate a comprehensive worldview that integrates diverse approaches to sustainability.
- PO 8: Ethics: Identify and demonstrate knowledge of ethical values in non-classroom activities, such as service learning, internships, and field work.
- PO 9: Individual and team work: An ability to function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- PO 10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give/receive clear instructions.
- PO 11: Project management and finance: An ability to demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- PO 12: Life-long learning: A recognition of the need for, and an ability to engage in life-long learning.

**Programme outcomes and Course outcomes mapping:**

Programme Outcomes	Course Outcomes					
	C01	C02	C03	C04	C05	C06
P01			✓	✓	✓	✓
P02		✓	✓	✓	✓	
P03		✓	✓	✓	✓	
P04		✓	✓	✓	✓	
P05						✓
P06						
P07						
P08						
P09					✓	
P010						
P011			✓	✓	✓	✓
P012						