

## SEMESTER S7

### SOFTWARE ENGINEERING

<b>Course Code</b>	<b>OECST723</b>	<b>CIE Marks</b>	40
<b>Teaching Hours/Week (L: T:P: R)</b>	3:0:0:0	<b>ESE Marks</b>	60
<b>Credits</b>	3	<b>Exam Hours</b>	2 Hrs. 30 Min.
<b>Prerequisites (if any)</b>	None	<b>Course Type</b>	Theory

#### Course Objectives:

1. To Provide fundamental knowledge in the Software Development Process including Software Development, Object Oriented Design, Project Management concepts and technology trends.
2. To enable the learners to apply state of the art industry practices in Software development.

### SYLLABUS

<b>Module No.</b>	<b>Syllabus Description</b>	<b>Contact Hours</b>
<b>1</b>	<p><b>Introduction to Software Engineering and Process Models</b> - Software engineering, Software characteristics and types, Layers of Software Engineering-Process, Methods, Tools and Quality focus. Software Process models – Waterfall, Prototype, Spiral, Incremental, Agile model – Values and Principles.</p> <p><b>Requirement engineering</b> - Functional, Non-functional, System and User requirements. Requirement elicitation techniques, Requirement validation, Feasibility analysis and its types, SRS document characteristics and its structure.</p> <p><i>Case study:</i> SRS for College Library Management Software</p>	<b>9</b>
<b>2</b>	<p><b>Software design</b> - Software architecture and its importance, Software architecture patterns: Component and Connector, Layered, Repository, Client-Server, Publish-Subscribe, Functional independence – Coupling and Cohesion</p> <p><i>Case study:</i> Ariane launch failure</p> <p><b>Object Oriented Software Design</b> - UML diagrams and relationships– Static and dynamic models, Class diagram, State diagram, Use case diagram,</p>	<b>10</b>

	<p>Sequence diagram</p> <p><i>Case Studies:</i> Voice mail system, ATM Example</p> <p><b>Software pattern</b> - Model View Controller, Creational Design Pattern types – Factory method, Abstract Factory method, Singleton method, Prototype method, Builder method. Structural Design Pattern and its types – Adapter, Bridge, Proxy, Composite, Decorator, Façade, Flyweight. Behavioral Design Pattern</p>	
3	<p><b>Coding, Testing and Maintenance:</b></p> <p><b>Coding guidelines</b> - Code review, Code walkthrough and Code inspection, Code debugging and its methods.</p> <p><b>Testing</b> - Unit testing , Integration testing, System testing and its types, Black box testing and White box testing, Regression testing</p> <p><b>Overview of DevOps and Code Management</b> - Code management, DevOps automation, Continuous Integration, Delivery, and Deployment (CI/CD/CD), <i>Case study</i> – Netflix.</p> <p><b>Software maintenance and its types-</b> Adaptive, Preventive, Corrective and Perfective maintenance. Boehm’s maintenance models (both legacy and non-legacy)</p>	10
4	<p><b>Software Project Management</b> - Project size metrics – LOC, Function points and Object points. Cost estimation using Basic COCOMO.</p> <p>Risk management: Risk and its types, Risk monitoring and management model</p> <p><b>Software Project Management</b> - Planning, Staffing, Organisational structures, Scheduling using Gantt chart. Software Configuration Management and its phases, Software Quality Management – ISO 9000, CMM, Six Sigma for software engineering.</p>	7

**Course Assessment Method**  
(CIE: 40 marks, ESE: 60 marks)

**Continuous Internal Evaluation Marks (CIE):**

Attendance	Assignment/ Microproject	Internal Examination-1 (Written)	Internal Examination- 2 (Written )	Total
5	15	10	10	40

**End Semester Examination Marks (ESE)**

*In Part A, all questions need to be answered and in Part B, each student can choose any one full question out of two questions*

Part A	Part B	Total
<ul style="list-style-type: none"> <li>• 2 Questions from each module.</li> <li>• Total of 8 Questions, each carrying 3 marks</li> </ul> <p align="center"><b>(8x3 =24 marks)</b></p>	<ul style="list-style-type: none"> <li>• Each question carries 9 marks.</li> <li>• Two questions will be given from each module, out of which 1 question should be answered.</li> <li>• Each question can have a maximum of 3 subdivisions.</li> </ul> <p align="center"><b>(4x9 = 36 marks)</b></p>	<b>60</b>

**Course Outcomes (COs)**

At the end of the course students should be able to:

Course Outcome		Bloom's Knowledge Level (KL)
<b>CO1</b>	Plan the system requirements and recommend a suitable software process model.	<b>K3</b>
<b>CO2</b>	Model various software patterns based on system requirements.	<b>K3</b>
<b>CO3</b>	Apply testing and maintenance strategies on the developed software product to enhance quality.	<b>K3</b>
<b>CO4</b>	Develop a software product based on cost, schedule and risk constraints.	<b>K3</b>

Note: K1- Remember, K2- Understand, K3- Apply, K4- Analyse, K5- Evaluate, K6- Create

**CO-PO Mapping Table (Mapping of Course Outcomes to Program Outcomes)**

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
<b>CO1</b>	3	3	3									3
<b>CO2</b>	3	3	3									3
<b>CO3</b>	3	3	3									3
<b>CO4</b>	3	3	3									3

Note: 1: Slight (Low), 2: Moderate (Medium), 3: Substantial (High), -: No Correlation

<b>Text Books</b>				
<b>Sl. No</b>	<b>Title of the Book</b>	<b>Name of the Author/s</b>	<b>Name of the Publisher</b>	<b>Edition and Year</b>
1	Software Engineering: A practitioner's approach	Roger S. Pressman	McGraw-Hill	8/e, 2014
2	Software Engineering	Ian Sommerville	Addison-Wesley	10/e, 2015
3	Design Patterns, Elements of Reusable Object Oriented Software	Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides	Pearson Education Addison-Wesley	1/e, 2009

<b>Reference Books</b>				
<b>Sl. No</b>	<b>Title of the Book</b>	<b>Name of the Author/s</b>	<b>Name of the Publisher</b>	<b>Edition and Year</b>
1	Pankaj Jalote's Software Engineering: With Open Source and GenAI	Pankaj Jalote	Wiley India	1/e, 2024
2	Software Engineering: A Primer	Waman S Jawadekar	Tata McGraw-Hill	1/e, 2008
3	Object-Oriented Modelling and Design with UML	Michael Blaha, James Rumbaugh	Pearson Education.	2/e, 2007
4	Software Engineering Foundations : A Software Science Perspective	Yingux Wang	Auerbach Publications	1/e, 2008
5	Object-Oriented Design and Patterns	Cay Horstmann	Wiley India	2/e, 2005
6	Engineering Software Products: An Introduction to Modern Software Engineering	Ian Sommerville	Pearson Education	1/e, 2020

**Video Links (NPTEL, SWAYAM...)**

<b>Module No.</b>	<b>Link ID</b>
<b>1</b>	<a href="https://www.youtube.com/watch?v=Z6f9ckEElsU">https://www.youtube.com/watch?v=Z6f9ckEElsU</a>
<b>2</b>	<a href="https://www.youtube.com/watch?v=1xUz1fp23TQ">https://www.youtube.com/watch?v=1xUz1fp23TQ</a>
<b>3</b>	<a href="http://digimat.in/nptel/courses/video/106105150/L01.html">http://digimat.in/nptel/courses/video/106105150/L01.html</a>
<b>4</b>	<a href="https://www.youtube.com/watch?v=v7KtPLhSMkU">https://www.youtube.com/watch?v=v7KtPLhSMkU</a>
<b>2</b>	<a href="https://archive.nptel.ac.in/courses/106/105/106105182/">https://archive.nptel.ac.in/courses/106/105/106105182/</a>