

24CSEN1021	SOFTWARE ENGINEERING	L	T	P	S	J	C
		2	0	4	0	0	4
Pre-requisite	24CSEN1011 - Object Oriented Programming						
Co-requisite	None						
Preferable Exposure	NIL						

#### Course Description:

Successful software development depends on an in-depth understanding of how the phases and supporting activities of the software development life cycle work together. Each phase of the life cycle contributes to a reliable, maintainable product that satisfies user requirements. This course presents modern software engineering techniques and examines the software life-cycle, including software specification, design, implementation, testing and maintenance.

#### Course Educational Objectives:

- To understand the Software Engineering Practice
- To understand the Software Engineering Process Models
- To understand Design Engineering
- To gain knowledge of the software testing
- To understand Software Process and Project Management

#### MODULE 1 INTRODUCTION TO SOFTWARE AND SOFTWARE ENGINEERING

12 Hrs

The Evolving Role of Software, Software: A Crisis on the Horizon and Software Myths, Software Engineering: A Layered Technology, Software Process Models, The Linear Sequential Model, The Prototyping Model, The RAD Model, Evolutionary Process Models, Component-Based Development, Process, Product and Process, Agile Process Model & Development.

Exemplar/Case Studies: Suggested FOSS TOOLS: Agile Tools -Jira, Study Library management system in line with software engineering aspects.

#### MODULE 2 REQUIREMENT ANALYSIS AND SPECIFICATION

12 Hrs

Requirement Analysis and Specification:

Understanding the Requirement, Requirement Modelling, Requirement Specification (SRS), Requirement Analysis and Requirement Elicitation, Requirement Engineering

Managing Software Projects:

Software Metrics (Process, Product and Project Metrics), Software Project Estimates, Software Project Planning (MS Project Tool), Project Scheduling & Tracking, Risk Analysis & Management (Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation).

Exemplar/Case Studies: Write SRS for Library management system , Suggested FOSS TOOLS: StartUML, SmartDraw, Modelio ,GanttProjector Project Libre

#### MODULE 3 SOFTWARE DESIGN

12 Hrs

Software Design: Outcome of a design process, cohesion and coupling, layered arrangement of modules, approaches to software design, function-oriented software design: overview of SA/SD methodology, structured analysis, DFDs, structured design, detailed design, design review, object-oriented software design: UML diagrams, use case modeling, unified process, OOD goodness criteria, user interface design, types of user interfaces, component-based GUI development. Web Application Design.

Exemplar/Case Studies: Study design of Library management system software,

Suggested FOSS TOOLS: PlantUML , SmartDraw

#### **MODULE 4 CODING AND TESTING**

**12 Hrs**

Coding and Testing: Coding standards and guidelines, code review, software documentation, unit testing, black-box testing, white-box testing, test coverage – code coverage, condition coverage, branch coverage, debugging, integration testing, system testing. Test Case, Test Suites Design, Testing Object Oriented Applications, Testing Web and Mobile Applications, Testing Tools (Win runner, Load runner).

Exemplar/Case Studies: Design testcases (Jenny and Pict) and coverage analysis related to Library management system

Suggested FOSS TOOLS: Junit, Cobertura, Selenium Testing (online Library management system application)

#### **MODULE 5 QUALITY ASSURANCE AND MANAGEMENT AND DEVOPS**

**12 Hrs**

Software reliability and Quality management: Software reliability, Statistical testing, software quality, ISO 9000, SEI CMM, PSP, Six sigma, CASE Tools, Software maintenance, Software reuse.

DevOps Importance and Benefits, DevOps Principles and Practices, 7 DevOps Lifecycle for Business Agility, DevOps and Continuous Testing, How to Choose Right DevOps Tools

Exemplar/Case Studies: Analysis of a project using PMBOK concepts and tools related to DevOps

#### **List of Experiments**

<b>S.no</b>	<b>Topic</b>	<b>Type</b>
1	Draft a System Requirements Specification	Experiment
2	Estimation of Cost and Schedule	Experiment
3	Design using Entity Relationship Diagram, Context flow diagram, DFD (Structural Modeling and Functional Modeling)	Experiment
4	Design using State Transition Diagrams (Behavioral Modeling)	Experiment
5	Design usingUML diagrams for OO Design	Experiment
6	Perform Black-box, White-box testing , Non-functional testing	Experiment
7	Use Tools for Version Control	Experiment
8	Identify the Software Project, Create Business Case, Arrive at a Problem Statement	Project
9	Analyze Stakeholder and User Description and Identify the appropriate Process Model	Project
10	Identify the Requirements, System Requirements, Functional Requirements, Non-Functional Requirements and develop a SRS Document	Project
11	Prepare Project Plan based on scope, Find Job roles and responsibilities, Calculate Project effort based on resources	Project
12	Prepare the Work, Breakdown Structure based on timelines, Risk Identification and Plan	Project
13	Design a System Architecture, Use Case Diagram, ER Diagram (Database)	Project
14	DFD Diagram (process) (till Level 1), Class Diagram (Applied For OOPS based Project)	Project
15	Interaction Diagrams, State chart and Activity Diagrams, State and Sequence Diagram, Deployment Diagram,	Project
16	Sample Frontend Design (UI/UX)	Project

17	Sample code implementation	Project
18	Master Test Plan, Test Case Design	Project
19	Manual Testing	Project
20	User Manual, Analysis of Costing, Effort and Resource	Project
21	Why Agile process models with DevOps are recommended in big Software companies. Justify with above project studies	Project

**Textbook(s):**

1. R. S. Pressman, Software Engineering: A Practitioner's Approach, 9th Edition, McGraw Hill Publications, , 2020 , ISBN: 9781259872976
2. R. Mall, Fundamentals of Software Engineering, 5th Edition, PHI Learning Publisher, , 2018 ,ISBN 978-9388028028

**Reference(s):**

1. Pankaj Jalote, Software Project Management in Practice, Addison-Wesley, 2015 ,ISBN: 9789332547940
2. NPTEL Software Engineering, ,[https://onlinecourses.nptel.ac.in/noc19\\_cs69/preview](https://onlinecourses.nptel.ac.in/noc19_cs69/preview)
3. Robert C. Martin, Clean Code: A Handbook of Agile Software Craftsmanship, 1st Edition, Pearson Publisher, , 2008 , ISBN: 978-0132350884

**Course Outcomes:**

1. Assess each module given the overall Software engineering practice. (L4)
2. Design an SRS using technical and non-technical requirements and complete requirement engineering task for project (L6)
3. Design and develop a software product in accordance with SE principles.(L6)
4. Apply testing techniques to various types of software projects. (L3)
5. Apply software development and operation management activities with DevOps (L3)

**Course Articulation Matrix:**

	POs																PSOs			
CO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	1	2	3	4
1	3	2	1		1				1			2					2			
2	3	2	2		1				1			2					1			
3	3	3	2		1				1			2					1			
4	3	3	2	1	3				2			2					1			
5	3	3	2	2	3				3			2					1			

3 – High, 2 – Medium & 1 – Low Correlation

**APPROVED IN MEETINGS HELD ON:**

BOS : 03-02-2024

Academic Council Number: 27

Academic Council : 06-07-2023

**SDG No(s). & Statement(s) :**

4 & Quality Education : Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.Ensure inclusive and equitable quality education and promote lifelong learning opportunities for all.

**SDG Justification(s):**

SDG 4: The modules and topics mentioned in this course are designed to ensure all-inclusive and thorough education with equity to all Software Programmers and always promote learning opportunities.